SKP Engineering College

Tiruvannamalai – 606611

A Course Material

on

Ad hoc and Sensor Networks

By

P.Bharathi Vikkiraman

Assistant Professor

Computer Science and Engineering Department

Quality Certificate

This is to Certify that the Electronic Study Material

Subject Code: CS6003

Subject Name: Ad hoc and Sensors Networks

Year/Sem:IV/VIII

Being prepared by me and it meets the knowledge requirement of the University curriculum.

Signature of the Author

Name: P.Bharathi Vikkiraman

Designation: Assistant Professor

This is to certify that the course material being prepared by Mr. P.Bharathi Vikkiraman is of the adequate quality. He has referred more than five books and one among they is from abroad author.

Signature of HD	Signature of the Principal
Name:Mr.R.Saravanakumar	Name: Dr.V.Subramania Bharathi
Seal:	Seal:

CS6003 AD HOC AND SENSOR NETWORKS L T P C 3 0 0 3

OBJECTIVES:

The student should be made to:

- Understand the design issues in ad hoc and sensor networks.
- Learn the different types of MAC protocols.
- Be familiar with different types of adhoc routing protocols.
- Be expose to the TCP issues in adhoc networks
- .Learn the architecture and protocols of wireless sensor networks.

UNIT I INTRODUCTION 9

Fundamentals of Wireless Communication Technology – The Electromagnetic Spectrum – Radio propagation Mechanisms – Characteristics of the Wireless Channel - mobile ad hoc networks (MANETs) and wireless sensor networks (WSNs) :concepts and architectures. Applications of Ad Hoc and Sensor Networks. Design Challenges in Ad hoc and Sensor Networks.

UNIT II MAC PROTOCOLS FOR AD HOC WIRELESS NETWOR 9

Issues in designing a MAC Protocol- Classification of MAC Protocols- Contention based protocolsContention based protocols with Reservation Mechanisms- Contention based protocols with Scheduling Mechanisms – Multi channel MAC-IEEE 802.11 106

UNIT III ROUTING PROTOCOLS AND TRANSPORT LAYER IN AD HOC WIRELESS NETWORKS 9

Issues in designing a routing and Transport Layer protocol for Ad hoc networks- proactive routing, reactive routing (on-demand), hybrid routing- Classification of Transport Layer solutions-TCP over Ad hoc wireless Networks.

UNIT IV WIRELESS SENSOR NETWORKS (WSNS) AND MAC PROTOCOLS 9

Single node architecture: hardware and software components of a sensor node - WSN Network architecture: typical network architectures-data relaying and aggregation strategies -MAC layer protocols: self-organizing, Hybrid TDMA/FDMA and CSMA based MAC- IEEE 802.15.4.

UNIT V WSN ROUTING, LOCALIZATION & QOS 9

Issues in WSN routing – OLSR- Localization – Indoor and Sensor Network Localization-absolute and relative localization, triangulation-QOS in WSN-Energy Efficient Design-Synchronization-Transport Layer issues.

TOTAL: 45 PERIODS

OUTCOMES:

- Upon completion of the course, the student should be able to:
- Explain the concepts, network architectures and applications of ad hoc and wireless sensor
- networks Analyze the protocol design issues of ad hoc and sensor networks
- Design routing protocols for ad hoc and wireless sensor networks with respect to some protocol
- design issues Evaluate the QoS related performance measurements of ad hoc and sensor network

TEXT BOOK:

1. C. Siva Ram Murthy, and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols ", Prentice Hall Professional Technical Reference, 2008.

REFERENCES:

1. Carlos De Morais Cordeiro, Dharma Prakash Agrawal “Ad Hoc & Sensor Networks: Theory and Applications”, World Scientific Publishing Company, 2006.

2. Feng Zhao and Leonides Guibas, "Wireless Sensor Networks", Elsevier Publication - 2002.

3. Holger Karl and Andreas Willig “Protocols and Architectures for Wireless Sensor Networks”, Wiley, 2005

4. Kazem Sohraby, Daniel Minoli, & Taieb Znati, “Wireless Sensor Networks-Technology, Protocols, and Applications”, John Wiley, 2007.

5. Anna Hac, “Wireless Sensor Network Designs”, John Wiley, 2003.

CONTENTS

Unit – I

INTRODUCTION

Part – A

1. What is an adhoc network? [CO1-L1]

Simply stating, a Mobile Ad hoc NETwork (MANET) is one that comes together as needed, not necessarily with any support from the existing Internet infrastructure or any other kind of fixed stations.

2. What is fading? List the different types of fading? [CO1-L1]

We can formalize this statement by defining an ad hoc network as an autonomous system of mobile hosts (also serving as routers) connected by wireless links, the union of which forms a communication network modeled in the form of an arbitrary graph.

3. Why ad hoc networks are needed? [CO1-L2]

As for the mode of operation, ad hoc networks are basically peer-to-peer multi-hop mobile wireless networks where information packets are transmitted in a store-andforward manner from a source to an arbitrary destination.

4. What are the challenging issues in ad hoc network maintenance? [CO1-L1]

I. Logical time of a link failure
II. The unique ID of the node that defined the new reference level
III. A reflection indicator bit
IV. A propagation ordering parameter.

5. What is hidden terminal problem? [CO1-L1]

The Location-Aided Routing (LAR) [Ko 1998] protocol exploits location information to limit the scope of route request flood employed in protocols such as AODV and DSR. Such location information can be obtained through GPS (Global Positioning System).

6. Distinguish between shadowing and reflection of signal propagation[CO1-L2]

If node S does not know a previous location of node D, then node S cannot reasonably determine the expected zone (the entire region that may potentially be occupied by the ad hoc network is assumed to be the expected zone). In this case, LAR reduces to the basic flooding algorithm

7. List the transmission impediments of wireless channel [CO1-L1]

It is an infrastructureless IP based network of mobile and wireless machine nodes connected with radio. In operation, the nodes of a MANET do not have a centralized administration mechanism. It is known for its routeable network properties where each node act as a “router” to forward the traffic to other specified node in the network.

8. State Shanon’s theorem. [CO1-L1]

When the wave travels through a medium, which contains many objects with dimensions small when compared to its wavelength, scattering occurs. The wave gets scattered into several weaker outgoing signals. In practice, objects such as street signs, lamp posts, and foliage cause scattering.

9. Differentiate an ad hoc network and a cellular network with respect to bandwidth usage and cost effectiveness [CO1-L1]

Infrared waves and waves in the EHF band (also known as millimeter waves) are used for shortrange communication.

They are widely used in television, VCR, and stereo remote controls.

- They are relatively directional and inexpensive to build.
- They cannot travel through obstacles, which is a light-like

10. List the applications of ad hoc network. [CO1-L2]

1. Military Application
2. Collaborative & Distributed computing
3. Emergency Operations
4. Wireless Mesh Network

11. Define path loss. [CO1-L1]

Path loss can be expressed as the ratio of the power of the transmitted signal to the power of the same signal received by the receiver, on a given path.

It is a function of the propagation distance.

Path loss is dependent on a number of factors such as the radio frequency used and the nature.

Since several of these factors (in particular, the terrain) cannot be the same everywhere, a single
model may not be enough. So, several models are required to describe the variety of transmission environments.

12. List the issues that affect the design deployment and performance of ad hoc wireless system. [CO1-L2]

The scenario of deployment has significance because the capability required for a mobile node varies with the environment in which it is used. The following are some of the different scenarios in which the deployment issues vary¬ widely.

13. How Addressing and service discovery? [CO1-L2]

Addressing & service discovery assume significance in ad hoc wireless network due to the absence of any centralised coordinator.

An address that is globally unique in the connected part of the ad hoc wireless network is required for a node in order to participate in communication. .

Auto-configuration of addresses is required to allocate non-duplicate addresses to the nodes

14. List the benefits when deployment of a commercial ad hoc wireless networks compared to wired network. [CO1-L2]

The deployment of a commercial ad hoc wireless network has the following benefits when compared to wired networks.

Low cost of deployment

Incremental deployment

15. List the propagation mechanism experienced by radio wave. [CO1-L1]

- Reflection
- Diffraction
- Scattering

16. List the characteristics of wireless channel. [CO1-L1]

1. Path Loss
2. Fading
3. Interference

17. Define Nyquist theorem. [CO1-L2]

TheNyquist theorem gives the maximum data rate possible on a channel. If B is the bandwidth of the channel (in Hz) and L is the number of discrete signal levels/voltage values used, then the maximum channel capacity C according to the Nyquist theorem is given

18. Define Doppler shift. [CO1-L2]

The Doppler shift is defined as the change/shift in the frequency of the received signal when the transmitter and the receiver are mobile with respect to each other. If they are moving toward each other, then the frequency of the received signal will be higher than that of the transmitted signal, and if they are moving away from each other, the frequency of the signal at the receiver will be lower than that at the transmitter.

19. Define *Choice of protocol*. [CO1-L1]

The choice of protocols at different layers of the protocol stack is to be done taking into consideration the deployment scenario.

A TDMA-based & insecure MAC protocol may not be the best suited compared to a CDMA-based MAC protocol for a military application.

20. Define Scalability. [CO1-L1]

The multicast routing protocol should be able to scale for a network with a large number of node

<u>Part – B</u>

1. What are the characteristics and features of ad hoc networks? [CO1-H1]

CHARACTERISTICS OF THE WIRELESS CHANNEL

1.Path Loss

Path loss can be expressed as the ratio of the power of the transmitted signal to the power of the same signal received by the receiver, on a given path.

It is a function of the propagation distance. Path loss is dependent on a number of factors such as the radio frequency used and the nature of the terrain. Since several of these factors (in particular, the terrain) cannot be the same everywhere, a single model may not be enough. So, several models are required to describe the variety of transmission environments there are two path loss models,

1. Free propagation model
2. Two ray model or two path model Free propagation model

The simplest path loss model in which there is a direct-path signal between the transmitter and the receiver, with no atmospheric attenuation or multipath components. The relationship between the transmitted power Pt and the received power Pr is given by Where Gt and Gr are the transmitter and receiver antenna gains,

respectively, in the direction from the transmitter to the receiver, d is the distance between the transmitter and receiver, and λ = c/f (is the wavelength of the signal) Two ray model The signal reaches the receiver through two paths, one a line-of sight path, and the other the path through which the reflected (or refracted, or scattered) wave is received. According to the two-path model, the received power is given by where Pt is the transmitted power, Gt and Gr represent the antenna gains at the transmitter and the receiver, respectively, d is the distance between the transmitter and receiver, and ht and hr are the heights of the transmitter and the receiver, respectively.

2. Fading

Fading refers to the fluctuations in signal strength when received at the receiver. Fading can be classified into two types: 1. Fast fading/small-scale fading 2. Slow fading/large-scale fading. Fast fading refers to the rapid fluctuations in the amplitude, phase, or multipath delays of the received signal, due to the interference between multiple versions (copies) of the same transmitted signal arriving at the receiver at slightly different times. The time between the reception of the first version of the signal and the last echoed signal is called delay spread. The multipath propagation of the transmitted signal, which causes fast fading. The multipath propagation of the transmitted signal, which causes fast fading, The multiple signal paths may sometimes add constructively or sometimes destructively at the receiver, causing a variation in the power level of the received signal. Slow fading occurs when objects that partially absorb the transmissions lie between the transmitter and receiver. Slow fading is so called because the duration of the fade may last for multiple seconds or minutes. Slow fading may occur when the receiver is inside a building and the radio wave must pass through the walls of a building, or when the receiver is temporarily shielded from the transmitter by a building. Slow fading is also referred to as shadow fading since the objects that cause the fade, which may be large buildings or other structures, block the direct transmission path from the transmitter to the receiver. Some common measures to overcome the fading effect are

1. Diversity
2. Adaptive modulation

Diversity Mechanism based on the fact that independent paths between the same transmitter and receiver nodes experience independent fading effects. By providing multiple logical channels between the transmitter and receiver, and sending parts of the signal over each channel, the error effects due to fading can be compensated.

1. Time diversity mechanisms aim at spreading the data over time so that the effects of burst errors are minimized.
2. Frequency diversity mechanisms spread the transmission over a wider frequency spectrum, or use multiple carriers for transmitting the information.
3. Space diversity involves the use of different physical transmission paths.

Adaptive Modulation Mechanisms The channel characteristics are estimated at the receiver and the estimates are sent by the receiver to the transmitter through a feedback channel. The transmitter adapts its transmissions based on the received channel estimates in order to counter the errors that could occur due to the characteristics of the channel. Adaptive techniques are usually very complex to implement.

3. Interference

Wireless transmissions have to counter interference from a wide variety of sources. Two main forms of interference are adjacent channel interference and co-channel interference. 1. Adjacent channel interference case, signals in nearby frequencies have components outside their allocated ranges. These components may interfere with on-going transmissions in the adjacent frequencies. It can be avoided by carefully introducing guard bands2 between the allocated frequency ranges. 2.Co-channel interference, sometimes also referred to as narrow-band interference, is due to other nearby systems the same transmission frequency. Narrow-band interference due to frequency reuse in cellular systems can be minimized with the use of multiuser detection. A guard band is a small frequency band used to separate two adjacent frequency bands in order to avoid interference between them. Multiuser detection is an effective approach used to combat the multiuser interference problems inherent in CDMA systems Inter-symbol interference Inter-symbol interference is another type of interference, where distortion in the received signal is caused by the temporal spreading and the consequent overlapping of individual pulses in the signal. When this temporal spreading of individual pulses (delay spread) goes above a certain limit (symbol detection time), the receiver becomes unable to reliably distinguish between changes of state in the signal, that is, the bit pattern interpreted by the receiver is not the same as that sent by the sender. Adaptive equalization is a commonly used technique for combating inter-symbol interference. Adaptive equalization involves mechanisms for gathering the dispersed symbol energy into its•original time interval. Complex digital processing algorithms are used in the equalization process. The main principle behind adaptive equalization is the estimation of the channel pulse response to periodically transmitted well-known bit patterns, known as training sequences. This would enable a receiver to determine the time dispersion of the channel and compensate accordingly.

4. Doppler Shift

The Doppler shift is defined as the change/shift in the frequency of the received signal when the transmitter and the receiver are mobile with respect to each other. If they are moving toward each other, then the frequency of the received signal will be higher than that of the transmitted signal, and if they are moving away from each other, the frequency of the signal at the receiver will be lower than that at the transmitter. The Doppler shift fd is given by•where v is the relative velocity between the transmitter and receiver, and λ is the wavelength of the signal.

5. Transmission Rate

Constraints Two important constraints that determine the maximum rate of data transmission on a channel are Nyquist's theorem and Shannon's theorem. Nyquist's Theorem The signaling speed of a transmitted signal denotes the number of times per second the¬ signal changes its value/voltage. The number of changes per second is measured in terms of baud. The baud rate is not the same as the bit rate/data rate of the signal since each signal value¬ may be used to convey multiple bits. . TheNyquist theorem gives the maximum data rate possible on a channel. If B is the¬ bandwidth of the channel (in Hz) and L is the number of discrete signal levels/voltage values used, then the maximum channel capacity C according to the Nyquist's theorem is given by Shannon's Theorem Noise lev¬el in the channel is represented by the SNR. It is the ratio of signal power (S) to noise power (N), specified in decibels, that is, SNR = 10 log10(S/N). Shannon was his theorem on the maximum data rate possible on a noisy channel. According¬ to Shannon's theorem, the maximum data rate C is given by where B is the bandwidth of the channel (in Hz).

APPLICATIONS OF AD HOC WIRELESS NETWORKS

1. Military Application

Ad hoc wireless networks can be very useful in establishing communication among a group of soldiers for tactical operations. Setting up of a fixed infrastructure for communication among group of soldiers in enemy territories or in inhospitable terrains may not be possible. In such a case, ad hoc wireless networks provide required communication mechanism quickly. Coordination of military objects moving at high speeds such as fleets of airplanes, warships. Ad hoc used in secure communication, compromise of other security threats and safety of personnel in the tactical operation. In military application multiple high- power transceivers, each with the ability to hop between different frequencies, with long life batteries www.rejinpaul.com The primary nature of the communication required in a military environment enforces certain important requirements on adhoc wireless networks namely, Reliability, Efficiency, Secure communication & Support for multicast routing.

2. Collaborative & Distributed computing

Ad hoc wireless network helps in collaborative computing, by establishing temporary communication infrastructure for quick communication with minimal configuration among a group of people in a conference. In distributed file sharing application reliability is of high importance which would be provided by ad hoc network. Other applications such as streaming of multimedia objects among participating nodes in ad hoc Wireless networks require support for soft real-time communication Devices used for such applications could typically be laptops with add -on wireless interface cards, enhanced personal digital assistants (PDAs) or mobile devices with high processing power

3. Emergency Operations

Ad hoc wireless networks are very useful in emergency operations such as search and rescue, crowd control and commando operations The major factors that favour ad hoc wireless networks for such tasks are self-configuration of the• system with minimal overhead, independent of fixed or centralised infrastructure, the freedom and flexibility of mobility, and unavailability of conventional communication infrastructure. In environments, where the conventional infrastructure based communication facilities aredestroyed due to a war or due to natural calamities, immediate deployment of adhoc wireless networks would be a good solution for co-ordinating rescue activities. They require minimum initial network configuration with very little or no delay

4. Wireless Mesh Network

Wireless mesh networks are ad hoc wireless network that are formed to provide an alternate communication infrastructure for mobile or fixed nodes/users, without the spectrum reuse constraint & requirement of network planning of cellular network. It provides many alternate paths for a data transfer session between a source & destination, resulting in quick reconfiguration of the path when the existing path fails due to node failure. Since the infrastructure built is in the form of small radio relaying devices, the investment requires in wireless mesh networks are much less than what is required for the cellular network counterpart. The possible deployment scenarios of wireless mesh networks include: residential zones, highways, business zones, important civilian regions and university campuses Wireless mesh networks should be capable of self-organization and maintenance. It operates at license-free ISM band around 2.4 GHz & 5 GHz. It is scaled well to provide support to large number of points. Major advantage is the support for a high data rate, quick& low cost of deployment, enhanced services, high scalability, easy extendibility, high availability & low cost per bit. Wireless Sensor Networks: Sensor networks are special category of Adhoc wireless network that are used to provide a wirelescommunication infrastructure among the sensors deployed in a specific application domain. Sensor nodes are tiny devices that have capability of sensing physical parameters processing the• data gathered, & communication to the monitoring system . www.rejinpaul.com The issue that make sensor network a distinct category of adhoc wireless network are the following:

1. Mobility of nodes : Mobility of nodes is not a mandatory requirement in sensor networks.¬ For example, the nodes used for periodic monitoring of soil properties are not required to be¬ mobile & the nodes that are fitted on the bodies of patients in a post -surgery ward of a hospital are designed to support limited or partial mobility In general, sensor networks need not in all cases be designed to support mobility of sensor¬ nodes.

2. Size of the network : The number of nodes in sensor network can be much larger than that in a typical ad hoc wireless network.

3. Density of deployment : The density of nodes in a sensor network varies with the domain of application. For example, Military applications require high availability of the network, making redundancy a high priority.

4. Power constraints : The power constraints in sensor networks are much more stringent than those in ad hoc¬ wireless networks. This is mainly because the sensor nodes are expected to operate in harsh environmental or geographical conditions, with

minimum or no human supervision and maintenance. In certain case, the recharging of the energy source is impossible.¬ Running such a network, with nodes powered by a battery source with limited energy,¬ demands very efficient protocol at network, data link, and physical layer. The power sources used in sensor networks can be classified into the following 3 categories

1. Replenishable Power source: The power source can be replaced when the existing source is fully drained.
2. Non-replenishable Power source: The power source cannot be replenished once the network has been deployed. The replacement of sensor node is the only solution.
3. Regenerative Power source

2. Explain path loss and fading in detail. [CO1-H1]

1.Path Loss

Path loss can be expressed as the ratio of the power of the transmitted signal to the power of the same signal received by the receiver, on a given path.

It is a function of the propagation distance. Path loss is dependent on a number of factors such as the radio frequency used and the nature of the terrain. Since several of these factors (in particular, the terrain) cannot be the same everywhere, a single model may not be enough. So, several models are required to describe the variety of transmission environments There are two path loss maodel,

1. Free propagation model
2. Two ray model or two path model Free propagation model

The simplest path loss model in which there is a direct-path signal between the transmitter and the receiver, with no atmospheric attenuation or multipath components. The relationship between the transmitted power Pt and the received power Pr is given by Where Gt and Gr are the transmitter and receiver antenna gains, respectively, in the direction from the transmitter to the receiver, d is the distance between the transmitter and receiver, and λ = c/f (is the wavelength of the signal) Two ray model The signal reaches the receiver through two paths, one a line-of sight path, and the other the path through which the reflected (or refracted, or scattered) wave is received. According to the two-path model, the received power is given by where Pt is the transmitted power, Gt and Gr represent the antenna gains at the transmitter and the receiver, respectively, d is the distance between the transmitter and receiver, and ht and hr are the heights of the transmitter and the receiver, respectively.

2. Fading

Fading refers to the fluctuations in signal strength when received at the receiver. Fading can be classified into two types: 1. Fast fading/small-scale fading 2. Slow fading/large-scale fading. Fast fading refers to the rapid fluctuations in the amplitude, phase, or multipath delays of the received signal, due to the interference between multiple versions (copies) of the same transmitted signal arriving at the receiver at slightly different times. The time between the reception of the first version of the signal and the last echoed signal is called delay spread. The multipath propagation of the transmitted signal, which causes fast fading. The multipath propagation of the transmitted signal, which causes fast fading, The multiple signal paths may sometimes add constructively or sometimes destructively at the receiver, causing a variation in the power level of the received signal. Slow fading occurs when objects that partially absorb the transmissions lie between the transmitter and receiver. Slow fading is so called because the duration of the fade may last for multiple seconds or minutes. Slow fading may occur when the receiver is inside a building and the radio wave must pass through the walls of a building, or when the receiver is temporarily shielded from the transmitter by a building. Slow fading is also referred to as shadow fading since the objects that cause the fade, which may be large buildings or other structures, block the direct transmission path from the transmitter to the receiver. Some common measures to overcome the fading effect are

1. Diversity
2. Adaptive modulation

Diversity Mechanism Based on the fact that independent paths between the same transmitter and receiver nodes experience independent fading effects. By providing multiple logical channels between the transmitter and receiver, and sending parts of the signal over each channel, the error effects due to fading can be compensated.

1. Time diversity mechanisms aim at spreading the data over time so that the effects of burst errors are minimized.
2. Frequency diversity mechanisms spread the transmission over a wider frequency spectrum, or use multiple carriers for transmitting the information.
3. Space diversity involves the use of different physical transmission paths.

Adaptive Modulation Mechanisms The channel characteristics are estimated at the receiver and the estimates are sent by the receiver to the transmitter through a feedback channel. The transmitter adapts its transmissions based on the received channel estimates in order to counter the errors that could occur due to the characteristics of the channel. Adaptive techniques are usually very complex to implement.

3. Explain the two main forms of interference, Doppler shift and Nyquist theorem. [CO1-H2]

4. Doppler Shift

The Doppler shift is defined as the change/shift in the frequency of the received signal when the transmitter and the receiver are mobile with respect to each other. If they are moving toward each other, then the frequency of the received signal will be higher than that of the transmitted signal, and if they are moving away from each other, the frequency of the signal at the receiver will be lower than that at the transmitter.

The Doppler shift *fd* is given by

where *v* is the relative velocity between the transmitter and receiver, and λ is the wavelength of the signal.

4. Explain the applications areas of ad hoc networks. [CO1-H1]

1. Military Application

- Ad hoc wireless networks can be very useful in establishing communication among a group of soldiers for tactical operations.
- Setting up of a fixed infrastructure for communication among group of soldiers in enemy territories or in inhospitable terrains may not be possible.
- In such a case, ad hoc wireless networks provide required communication mechanism quickly.
- Coordination of military objects moving at high speeds such as fleets of airplanes, warships.
- Ad hoc used in secure communication, compromise of other security threats and safety of personnel in the tactical operation.
- In military application multiple high- power transceivers, each with the ability to hop between different frequencies, with long life batteries
- The primary nature of the communication required in a military environment enforces certain important requirements on adhoc wireless networks namely, Reliability, Efficiency, Secure communication & Support for multicast routing.

2. Collaborative & Distributed computing

- Ad hoc wireless network helps in collaborative computing, by establishing temporary communication infrastructure for quick communication with minimal configuration among a group of people in a conference.
- In distributed file sharing application reliability is of high importance which would be provided by ad hoc network.
- Other applications such as streaming of multimedia objects among participating nodes in ad hoc
- Wireless networks require support for soft real-time communication
- Devices used for such applications could typically be laptops with add -on wireless interface cards, enhanced personal digital assistants (PDAs) or mobile devices with high processing power

3. Emergency Operations

- Ad hoc wireless networks are very useful in emergency operations such as search and rescue, crowd control and commando operations
- The major factors that favour ad hoc wireless networks for such tasks are self-configuration of the system with minimal overhead, independent of fixed or centralised infrastructure, the freedom and flexibility of mobility, and unavailability of conventional communication infrastructure.
- In environments, where the conventional infrastructure based communication facilities are destroyed due to a war or due to natural calamities, immediate deployment of adhoc wireless networks would be a good solution for co-ordinating rescue activities.
- They require minimum initial network configuration with very little or no delay

4. Wireless Mesh Network

- Wireless mesh networks are ad hoc wireless network that are formed to provide an alternate communication infrastructure for mobile or fixed nodes/users, without the spectrum reuse constraint & requirement of network planning of cellular network.
- It provides many alternate paths for a data transfer session between a source & destination, resulting in quick reconfiguration of the path when the existing path fails due to node failure.
- Since the infrastructure built is in the form of small radio relaying devices, the investment required in wireless mesh networks is much less than what is required for the cellular network counterpart.

- The possible deployment scenarios of wireless mesh networks include: residential zones, highways, business zones, important civilian regions and university campuses
- Wireless mesh networks should be capable of self-organization and maintenance.
- It operates at license-free ISM band around 2.4 GHz & 5 GHz.
- It is scaled well to provide support to large number of points.
- Major advantage is the support for a high data rate, quick & low cost of deployment, enhanced services, high scalability, easy extendibility, high availability & low cost per bit.

Wireless Sensor Networks:

- Sensor networks are special category of Adhoc wireless network that are used to provide a wireless communication infrastructure among the sensors deployed in a specific application domain.

1. ***Mobility of nodes*** :

□ Mobility of nodes is not a mandatory requirement in sensor networks.

□ For example, the nodes used for periodic monitoring of soil properties are not required to be mobile & the nodes that are fitted on the bodies of patients in a post - surgery ward of a hospital are designed to support limited or partial mobility

□ In general, sensor networks need not in all cases be designed to support mobility of sensor nodes.

2. ***Size of the network*** :

The number of nodes in sensor network can be much larger than that in a typical ad hoc wireless network.

3. ***Density of deployment*** :

The density of nodes in a sensor network varies with the domain of application. For example, Military applications require high availability of the network, making redundancy a high priority.

4. ***Power constraints*** :

- The power constraints in sensor networks are much more stringent than those in ad hoc wireless networks. This is mainly because the sensor nodes are expected to operate in harsh environmental or geographical conditions, with minimum or no human supervision and maintenance.

- In certain case, the recharging of the energy source is impossible.
- Running such a network, with nodes powered by a battery source with limited energy, demands very efficient protocol at network, data link, and physical layer.
- The power sources used in sensor networks can be classified into the following 3 categories:

1. Replenishable Power source: The power source can be replaced when the existing source is fully drained.

2. Non-replenishable Power source: The power source cannot be replenished once the network has been deployed. The replacement of sensor node is the only solution.

3. Regenerative Power source: Here, Power source employed in sensor network have the capability of regenerating power from the physical parameter under measurement.

5. Data / Information fusion :

□ Data fusion refers to the aggregation of multiple packets into one before relaying it.

□ Data fusion mainly aims at reducing the bandwidth consumed by redundant headers of the packets and reducing the media access delay involved in transmitting multiple packets.

6. Traffic Distribution :

- □ The communication traffic pattern varies with the domain of application in sensor networks.
- □ For example, the environmental sensing application generates short periodic packets indicating the status of the environmental parameter under observation to a central monitoring station.
- □ This kind of traffic requires low bandwidth.
- □ Ad hoc wireless networks generally carry user traffic such as digitized & packetized voice stream or data traffic, which demands higher bandwidth.

5. Explain the ISSUES IN AD HOC WIRELESS NETWORKS. [CO1-H2]

ISSUES IN AD HOC WIRELESS NETWORKS

The major issues that affect the design, deployment, & performance of an ad hoc wireless network system are:

1. Medium Access Scheme.
2. Transport Layer Protocol.
3. Routing.
4. Multicasting.
5. Energy Management.
6. Self-Organisation.
7. Security.
8. Addressing & Service discovery.
9. Deployment considerations.
10. Scalability.
11. Pricing Scheme.
12. Quality of Service Provisioning

1. Medium Access Scheme

The primary responsibility of a Medium Access Control (MAC) protocol in adhoc wireless networks is the distributed arbitration for the shared channel for transmission of packets. The major issues to be considered in designing a MAC protocol for adhoc wireless networks are as follows:

- ***Distributed Operation:***
- □ The ad hoc wireless networks need to operate in environments where no centralized coordination is possible.

- □ The MAC protocol design should be fully distributed involving minimum control overhead.

Synchronization:

- The MAC protocol design should take into account the requirement of time synchronization.

- Synchronization is mandatory for TDMA-based systems for management of transmission and reception slots.

Hidden Terminals:

Hidden terminals are nodes that are hidden(or not reachable) from the sender of a data transmission session, but are reachable to the receiver of the session.

Exposed terminals:

Exposed terminals, the nodes that are in the transmission range of the sender of an on –going session, are prevented from making a transmission.

Throughput: The MAC protocol employed in adhoc wireless networks should attempt to maximize the throughput of the system.

The important considerations for throughput enhancement are
- Minimizing the occurrence of collisions.
- Maximizing channel utilization and
- Minimizing control overhead.

Access delay:

The average delay that any packet experiences to get transmitted. The MAC protocol should attempt to minimize the delay.

Fairness*:*

Fairness refers to the ability of the MAC protocol to provide an equal share or weighted share of the bandwidth to all competing nodes. Fairness can be either *node-based* or *flow-based.*

Real-time Traffic support**:**

In a contention-based channel access environment, without any central coordination, with limited bandwidth, and with location-dependent contention, supporting time-sensitive traffic such as voice, video, and real-time data requires explicit support from the MAC protocol.

Resource reservation**:**

The provisioning of QoS defined by parameters such as bandwidth, delay, and jitter requires reservation of resources such as *bandwidth, buffer space*, and *processing power*.

Ability to measure resource availability:

In order to handle the resources such as bandwidth efficiently and perform call admission control based on their availability, the MAC protocol should be able to provide an estimation of resource availability at every node. This can also be used for making *cogestion control decisions*.

Capability for power control:

□ The transmission power control reduces the energy consumption at the nodes, causes a decrease in interference at neighboring nodes, and increases frequency reuse.

Adaptive rate control:

This refers to the variation in the data bit rate achieved over a channel.
A MAC protocol that has adaptive rate control can make use of a high data rate when the sender and receiver are nearby & adaptively reduce the data rate as they move away from each other.

2. Routing

The responsibilities of a routing protocol include exchanging the route information; finding a feasible path to a destination. The major challenges that a routing protocol faces are as follows:

Mobility :

The Mobility of nodes results in frequent path breaks, packet collisions, transient loops, stale routing information, and difficulty in resource reservation.

Bandwidth constraint :

Since the channel is shared by all nodes in the broadcast region, the bandwidth available per wireless link depends on the number of nodes & traffic they handle.

Error-prone and shared channel : The Bit Error Rate (BER) in a wireless channel is very high [10-5 to 10 -3] compared to that in its wired counterparts [10-12 to 10-9]. Consideration of the state of the wireless link, signal-to-noise ratio, and path loss for routing in ad hoc wireless networks can improve the efficiency of the routing protocol.

Location-dependent contention :

The load on the wireless channel varies with the number of nodes present in a given geographical region.
This makes the contention for the channel high when the number of nodes increases. The high contention for the channel results in a high number of collisions & a subsequent wastage of bandwidth.

Other resource constraints :

The constraints on resources such as computing power, battery power, and buffer storage also limit the capability of a routing protocol. The major requirements of a routing protocol in adhoc wireless networks are the following.
1. Minimum route acquisition delay 2. Quick route reconfiguration
3. Loop-free routing 4. Distributed routing approach
5. Minimum control overhead 6. Scalability
7. Provisioning of QoS 8. Support for time-sensitive traffic:
9. Security and privacy

3. Multicasting:

It plays important role in emergency search & rescue operations & in military communication. Use of single link connectivity among the nodes in a multicast group results in a tree-shaped multicast routing topology. Such a tree-shaped topology provides high multicast efficiency, with low packet delivery ratio due to the frequency tree breaks. The major issues in designing multicast routing protocols are as follows:

Robustness :

The multicast routing protocol must be able to recover & reconfigure quickly from potential mobility- induced link breaks thus making it suitable for use in high dynamic environments.

Efficiency :

A multicast protocol should make a minimum number of transmissions to deliver a data packet to all the group members.

Control overhead :

The scarce bandwidth availability in ad hoc wireless networks demands minimal control overhead for the multicast session.

Quality of Service : QoS support is essential in multicast routing because, in most cases, the data transferred in a multicast session is time-sensitive.

Efficient group management : Group management refers to the process of accepting multicast session members and maintaining the connectivity among them until the session expires.

Scalability : The multicast routing protocol should be able to scale for a network with a large number of node

Security :

□ Authentication of session members and prevention of non-members from gaining unauthorized information play a major role in military communications.

Transport Layer Protocol

The main objectives of the transport layer protocols include:

Setting up & maintaining end-to-end connections, Reliable end-to-end delivery of packets, Flow control & Congestion control.

Examples of some transport layers protocols are,

a. UDP (User Datagram Protocol) :

It is an unreliable connectionless transport layer protocol. It neither performs flow control & congestion control. It do not take into account the current network status such as congestion at the intermediate links, the rate of collision, or other similar factors affecting the network throughput.

b. TCP (Transmission Control Protocol):

It is a reliable connection-oriented transport layer protocol. It performs flow control & congestion control. Here performance degradation arises due to frequent path breaks, presence of stale routing information, high channel error rate, and frequent network partitions.

5. Pricing Scheme

Assume that an optimal route from node A to node B passes through node C, & node C is not powered on.

Then node A will have to set up a costlier & non-optimal route to B.

The non-optimal path consumes more resources & affects the throughput of the system.
As the intermediate nodes in a path that relay the data packets expend their resources such as battery charge & computing power, they should be properly compensated.

Hence, pricing schemes that incorporate service compensation or service reimbursement are
required.

6. Quality of Service Provisioning (QoS)

- □ QoS is the performance level of services offered by a service provider or a network to the user.
- □ QoS provisioning often requires ,Negotiation between host & the network.
- □ Resource reservation schemes.
- □ Priority scheduling &
- □ Call admission control.

QoS parameters :

Applications Corresponding QoS parameter
1. Multimedia application - 1. Bandwidth & Delay
2. Military application - 2.Security & Reliability
3. Defense application - 3.Finding trustworthy intermediate hosts & routing
4. Emergency search and - 4 .Availability
rescue operations
5. Hybrid wireless network - 5.Maximum available link life, delay, bandwidth & channel utilization.

QoS-aware routing : Finding the path is the first step toward a QoS-aware routing protocol. The parameters that can be considered for routing decisions are, Network throughput, Packet delivery ratio, Reliability, Delay, Delay jitter, Packet loss rate Bit error rate, Path loss.

QoS framework :

I. A framework for QoS is a complete system that attempts to provide the promised services to
each user or application.

II. The key component of QoS framework is a QoS service model which defines the way user
requirements are served.

7. Self-Organization

One very important property that an ad hoc wireless network should exhibit is organizing &

maintaining the network by itself.

The major activities that an ad hoc wireless network is required to perform for self-organization are, Neighbour discovery, Topology organization & Topology reorganization (updating topology information)

8. Security

Security is an important issue in ad hoc wireless network as the information can be hacked. Attacks against network are of 2 types :

I. *Passive attack* → Made by malicious node to obtain information transacted in the network without disrupting the operation.

Further active attacks are of 2 types :

1. *External attack*: The active attacks that are executed by nodes outside the network.
2. *Internal attack:* The active attacks that are performed by nodes belonging to the same network.

The major security threats that exist in ad hoc wireless networks are as follows :

Denial of service

The attack affected by making the network resource unavailable for service to other nodes, either by consuming the bandwidth or by overloading the system.

Resource consumption

The scarce availability of resources in ad hoc wireless network makes it an easy target for internal attacks, particularly aiming at consuming resources available in the network.

The major types of resource consumption attacks are,

Energy depletion :

1. Highly constrained by the energy source
2. Aimed at depleting the battery power of critical nodes.

□*Buffer overflow* :

1. Carried out either by filling the routing table with unwanted routing entries or by consuming the data packet buffer space with unwanted data.
2. Lead to a large number of data packets being dropped, leading to the loss of criticalinformation.

Host impersonation

A compromised internal node can act as another node and respond with appropriate control packets to create wrong route entries, and can terminate the traffic meant for the intended destination node.

Information disclosure A compromised node can act as an informer by deliberate disclosure of confidential information to unauthorized nodes.

Interference

A common attack in defense applications to jam the wireless communication by creating a wide spectrum noise.

9. Addressing and service discovery

Addressing & service discovery assume significance in ad hoc wireless network due to the absence of any centralised coordinator. An address that is globally unique in the connected part of the ad hoc wireless network is required for a node in order to participate in communication. Auto-configuration of addresses is required to allocate non-duplicate addresses to the nodes.

□ Energy management is defined as the process of managing the sources & consumers of energy in a node or in the network for enhancing the lifetime of a network.

□ Features of energy management are :

→ shaping the energy discharge pattern of a node’s battery to enhance battery life.
→ finding routes that consumes minimum energy.
→Using distributed scheduling schemes to improve battery life.
→Handling the processor & interface devices to minimize power consumption.

□ Energy management can be classified into the following categories :

Transmission power management **:**

□ The power consumed by the Radio Frequency (RF) module of a mobile node is determined by several factors such as

* The state of operation.
* The transmission power and
* The technology used for the RF circuitry.

The state of operation refers to transmit, receive, and sleep modes of the operation.

The transmission power is determined by Reach ability requirement of the network, Routing protocol and MAC protocol employed.

Battery energy management : The battery management is aimed at extending the battery life of a node by taking advantage of its chemical properties, discharge patterns, and by the selection of a battery from a set of batteries that is available for redundancy.

Processor power management :

The clock speed and the number of instructions executed per unit time are some of the processor parameters that affect power consumption.
The CPU can be put into different power saving modes during low processing load conditions.
The CPU power can be completely turned off if the machines are idle for a long time. In such a cases, interrupts can be used to turn on the CPU upon detection of user interaction or other events.

Devices power management :

Intelligent device management can reduce power consumption of a mobile node significantly.
This can be done by the operating system(OS) by selectively powering down interface devices that are not used or by putting devices into different power saving modes, depending on their usage.

11. Scalability

□ Scalability is the ability of the routing protocol to scale well in a network with a large number of nodes.

□ It requires minimization of control overhead & adaptation of the routing protocol to the network size.

The deployment of a commercial ad hoc wireless network has the following benefits when compared to wired networks.

Low cost of deployment :

The use of multi-hop wireless relaying eliminates the requirement of cables & maintenance in deployment of communication infrastructure. The cost involved is much lower than that of wired networks.

Incremental deployment : Deployment can be performed incrementally over geographical regions of the city.

The deployed part of the network starts functioning immediately after the minimum configuration is done.

Short deployment time :
Compared to wired networks, the deployment time is considerably less due to the absence of any wired links.

Reconfigurability :

The cost involved in reconfiguring a wired network covering a Metropolitan Area Network(MAN) is very high compared to that of an ad hoc wireless network covering the same service area. The following are the major issues to be considered in deploying an ad hoc wireless network :

a) *Scenario of deployment* :
The scenario of deployment has significance because the capability required for a mobile node varies with the environment in which it is used.
The following are some of the different scenarios in which the deployment issues vary widely :

1. *Military deployment :*

- It can be either, Data-centric network : Handle a different pattern of data traffic & can be partially comprised of static nodes. Eg : a wireless sensor network.
- User-centric network: Consists of highly mobile nodes with or without any support from any infrastructure.
- Eg :soldiers or armored vehicles carrying soldiers equipped with wireless communication devices.

2. *Emergency operations deployment* :
Demands a quick deployment of rescue personnel equipped with hand-held communication equipment.

- The network should provide support for time-sensitive traffic such as voice & video.
- Short data messaging can be used in case the resource constraints do not permit voice communication.

3. *Commercial wide-area deployment :* Eg : wireless mesh networks.

3. Home network deployment :

Deployment needs to consider the limited range of the devices that are to be connected by the network.
Eg : short transmission range avoid network patitions.

b) ***Required longevity of network :***
If the network is required for a short while, battery-powered mobile nodes can be used. If the connectivity is required for a longer duration of time, fixed radio relaying equipment with regenerative power sources can be deployed.

c) ***Area of coverage :*** Determined by the nature of application for which the network is set up.
Eg : the home area network is limited to the surroundings of a home. The mobile nodes' capabilities such as the transmission range & associated hardware, software, & power source should match the area of coverage required.
d) ***Service availability** :*
Defined as the ability of an ad hoc wireless network to provide service even with the failure of certain nodes. Has significance in a Fully mobile ad hoc wireless network used for tactical communication & in partially fixed ad hoc wireless network used in commercial communication infrastructure such as wireless mesh networks.
e) ***Operational integration with other infrastructure :***
Considered for improving the performance or gathering additional information, or for providing better QoS.

In military environment, integration of ad hoc wireless networks with satellite networks or unmanned aerial vehicles (UAVs) improves the capability of the ad hoc wireless networks.

f) ***Choice of protocol :***
The choice of protocols at different layers of the protocol stack is to be done taking into consideration the deployment scenario. A TDMA-based & insecure MAC protocol may not be the best suited compared to a CDMA-based MAC protocol for a military application.

6. Explain the design issues in adhoc network? [CO1-H2]

FUNDAMENTALS OF WIRELESS COMMUNICATION TECHNOLOGY
It focuses on wireless networks, where electromagnetic radio waves are used for communication (exchange of information).
In the following sections, the various characteristics of radio propagation are first discussed.

Some of the important signal modulation mechanisms, multiple access techniques, and error control mechanisms are then described. A familiarity with all these fundamental aspects of wireless transmission is essential for understanding the issues involved in the design of wireless networks.

II.THE ELECTROMAGNETIC SPECTRUM

Wireless communication is based on the principle of broadcast and reception of electromagnetic waves.

These waves can be characterized by their frequency (f) or their wavelength (λ).

Frequency is the number of cycles (oscillations) per second of the wave and is measured in Hertz (Hz)

wavelength is the distance between two consecutive maxima or minima in the wave.

The **speed of propagation** of these waves (c) varies from medium to medium,

C= λ * f

where *c* is the speed of light (3 × 108*m*/*s*), *f* is the frequency of the wave in Hz, and λ is its wavelength in meters.

A pictographic view of the electromagnetic spectrum is given in Figure 1.1.

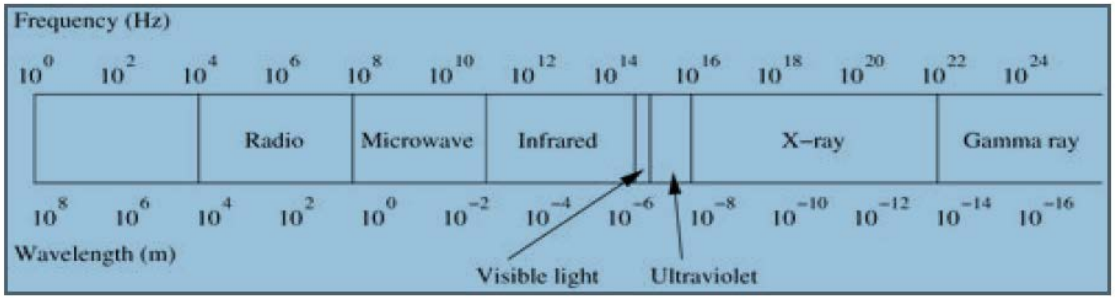


Figure 1.1. The electromagnetic spectrum

1) Radio waves

Easy to generate and are widely used for both **indoor and outdoor communication** due to properties such as their ability to pass through buildings and ability to travel long distances.

radio transmission is omnidirectional

At low frequencies the waves can pass through obstacles easily, but their power falls with an inverse-squared relation with respect to the distance.

The higher frequency waves are more prone to absorption by rain drops, and they get reflected by obstacles.

2) Propagation waves

VLF, LF, and MF bands the propagation of waves, also called as ground waves, follows the curvature of the Earth.

The maximum transmission ranges of these waves are of the order of a few hundred kilometers.

They are used for low bandwidth transmissions such as **amplitude modulated (AM) radio broadcasting.**

The **HF and VHF band transmissions** are absorbed by the atmosphere near the Earth's surface. However, a portion of the radiation, called the **sky wave** radiates outward and upward to the ionosphere in the upper atmosphere.

A powerful sky wave may get reflected several times between the Earth and the ionosphere.

Sky waves are used by amateur ham radio operators and for military communication.

3) Microwave

Microwave transmissions (in the SHF band) tend to **travel in straight lines** and hence can be narrowly focused.

Microwaves were widely used for **long-distance telephony,** before they got replaced by fiber optics.

They are also widely used for **mobile phones and television transmission**. Since the energy is concentrated

Higher signal-to-noise ratio. SNR is the ratio of the signal power to the noise power on a transmission medium, and is used to categorize the quality of a transmission.

The number of repeaters required is a function of the microwave transmission tower height.

4) Infrared waves

Infrared waves and waves in the EHF band (also known as millimeter waves) are used for **short-range communication**.

They are widely used in **television, VCR, and stereo remote controls**.

They are relatively directional and inexpensive to build.

They cannot travel through obstacles, which is a light-like

5) Visible light

The visible light part of the spectrum is just after the infrared portion.

Unguided optical signaling using visible light provides very high bandwidth at a very low cost.

Recent applications of light-wave transmissions involve the use of lasers to **connect LANs on two buildings through roof-top antennas.**

But the main disadvantage here is that it is very difficult to focus a very narrow unidirectional laser beam, which limits the maximum distance between the transmitter and receiver.

Unit -II

MAC PROTOCOLS FOR AD HOC WIRELESS NETWOR

Part –A

1. List the design goals of MAC protocol for ad- hoc networks. [CO2-L1]

- The operation of a protocol should be distributed
- The protocol should provide QoS support for real-time traffic
- The access delay, which refers to the average delay experienced by any packet to get transmitted,
- must be kept low
- The available bandwidth must be utilized efficiently
- The protocol should ensure fair allocation of bandwidth to nodes

2. List the issues of designing a MAC protocol for ad hoc networks. [CO2-L2]

- The protocol must be scalable to large networks

- It should have power control mechanisms in order to efficiently manage energy consumption of the nodes
- The protocol should have mechanisms for adaptive data rate control
- It should try to use directional antennas which can provide advantages such as reduced interference,
- increased spectrum reuse, and reduced power consumption
- The protocol should provide time synchronization among nodes

3. What are the classifications of MAC protocol [CO2-L1]

Ad hoc network MAC protocols can be classified into three basic types:
i. Contention-based protocols
ii. Contention-based protocols with reservation mechanisms
iii. Contention-based protocols with scheduling mechanisms
iv. Other MAC protocols

4. What are the effects of exposed terminal problem in wireless networks? [CO2-L1]

Sender-initiated protocols: Packet transmissions are initiated by the sender node.
Single-channel sender-initiated protocols: A node that wins the contention to the channel can make use of the entire bandwidth.
Multichannel sender-initiated protocols: The available bandwidth is divided into multiple channels.
Receiver-initiated protocols**:** The receiver node initiates the contention resolution protocol.

5. What are the advantages of directional antennas of MMAC over MACAW? [CO2-L2]

- Node scheduling is done in a manner so that all nodes are treated fairly and no node is starved of bandwidth.
- Scheduling-based schemes are also used for enforcing priorities among flows whose packets are queued at nodes.
- Some scheduling schemes also consider battery characteristics

6. What are the mechanisms used in MAC layer? [CO2-L2]

- If it is ready to transmit, the sender node respond by sending a DATA packet

- Thus data transmission in MACA-BI occurs through a two-way handshake mechanism
- The efficiency of the MACA-BI scheme is mainly dependent on the ability of the receiver node to predict accurately the arrival rates of traffic at the sender nodes

7. What are the differences between HRMA and SRMA? [CO2-L1]

SRAM (static **RAM**) is **random access memory** (**RAM**) that retains data bits in its memory as long as power is being supplied. Unlike dynamic **RAM** (**DRAM**), which stores bits in cells consisting of a capacitor and a transistor, **SRAM** does not have to be periodically refreshed

8. List the five phases of FPRP. [CO2-L1]

- CGSR is a hierarchical routing scheme which enables partial coordination between nodes by electing cluster-heads.
- Better bandwidth utilization is possible.
- Easy to implement priority scheduling schemes with token scheduling and gateway code scheduling.

9. List any two needs of real- time MAC protocol. [CO2-L2]

- These protocols are extensions of the wired network routing protocols
- They maintain the global topology information in the form of tables at every node
- Tables are updated frequently in order to maintain consistent and accurate network state information
- Ex: Destination sequenced distance vector routing protocol (DSDV), wireless routing protocol (WRP), source-tree adaptive routing protocol (STAR) and cluster-head gateway switch routing protocol (CGSR).

10. Compare the efficiency of the packet queuing mechanism adopted in MACA and MACAW[CO2-L2]

Takes multiple NDPUs. It is done either when the local topology changes significantly or when an incremental update requires more than a single NDPU.

Table updates are initiated by a destination with a new sequence number which is always greater than the previous one.

11. What do you mean by contention based protocols? [CO2-L1]

A **contention**-**based protocol** (CBP) is a communications **protocol** for operating wireless telecommunication equipment that allows many users to use the same radio channel without pre-coordination. The "listen before talk" operating procedure in IEEE 802.11 is the most well known **contention**-**based protocol**.

12. Give the classification of contention based protocol [CO2-L2]

Sender-initiated protocols: Packet transmissions are initiated by the sender node. Single-channel sender-initiated protocols: A node that wins the contention to the channel can make use of the entire bandwidth.

• Multichannel sender-initiated protocols: The available bandwidth is divided into multiple channels.

• Receiver-initiated protocols: The receiver node initiates the contention resolution protocol. Contention-based protocols with reservation mechanisms

• Synchronous protocols: All nodes need to be synchronized. Global time synchronization is difficult to achieve.

• Asynchronous protocols: These protocols use relative time information for effecting reservations.

13. Give the classifications of MAC protocols. [CO2-L2]

i. Contention-based protocols
ii. Contention-based protocols with reservation mechanisms
iii. Contention-based protocols with scheduling mechanisms

14. List the main issues in designing a MAC protocol for ad hoc wireless networks. [CO2-L1]

- The main issues need to be addressed while designing a MAC protocol for ad hoc wireless networks:
- Bandwidth efficiency is defined at the ratio of the bandwidth used for actual data transmission to the total available bandwidth. The MAC protocol for ad-hoc networks should maximize it.
- Quality of service support is essential for time-critical applications. The MAC protocol for ad-hoc networks should consider the constraint of ad-hoc networks.

- Synchronization can be achieved by exchange of control packets.

15. What do you mean by FAMA? [CO2-L1]

Floor acquisition Multiple Access Protocols (FAMA)
Based on a channel access discipline which consists of a carrier-sensing operation and a collision-avoidance dialog between the sender and the intended receiver of a packet.
Floor acquisition refers to the process of gaining control of the channel. At any time only one node is assigned to use the channel.
Carrier-sensing by the sender, followed by the RTS-CTS control packet exchange, enables the protocol to perform as efficiently as MACA

16. What do you mean by contention based protocols with reservation mechanism? [CO2-L1]

- RTS-CTS exchange with no carrier-sensing uses the ALOHA protocol for transmitting RTS packets.
- RTS-CTS exchange with non-persistent carrier-sensing uses non-persistent CSMA for the same purpose.

17. What do you mean by contention based protocols with scheduling mechanism? [CO2-L2]

- Contention occurs during the resource (bandwidth) reservation phase.
- Once the bandwidth is reserved, the node gets exclusive access to the reserved bandwidth.
- QoS support can be provided for real-time traffic.

18. What do you mean by D- PRMA? [CO2-L2]

- It extends the centralized packet reservation multiple access (PRMA) scheme into a distributed scheme that can be used in ad hoc wireless networks.
- PRMA was designed in a wireless LAN with a base station.
- D-PRMA extends PRMA protocol in a wireless LAN.
- D-PRMA is a TDMA-based scheme. The channel is divided into fixed- and equal-sized frames along the time axis.

19. What are the disadvantages of the binary exponential back off mechanism used in MACA? How are they overcome in MACAW? [CO2-L1]

- By eliminating the need for the RTS packet it reduces the number of control packets used in the MACA protocol which uses the three-way handshake mechanism.
- Media Access with Reduced Handshake (MARCH) is a receiver-initiated protocol.

20. What do you mean by BTMA? [CO2-L2]

The transmission channel is split into two:

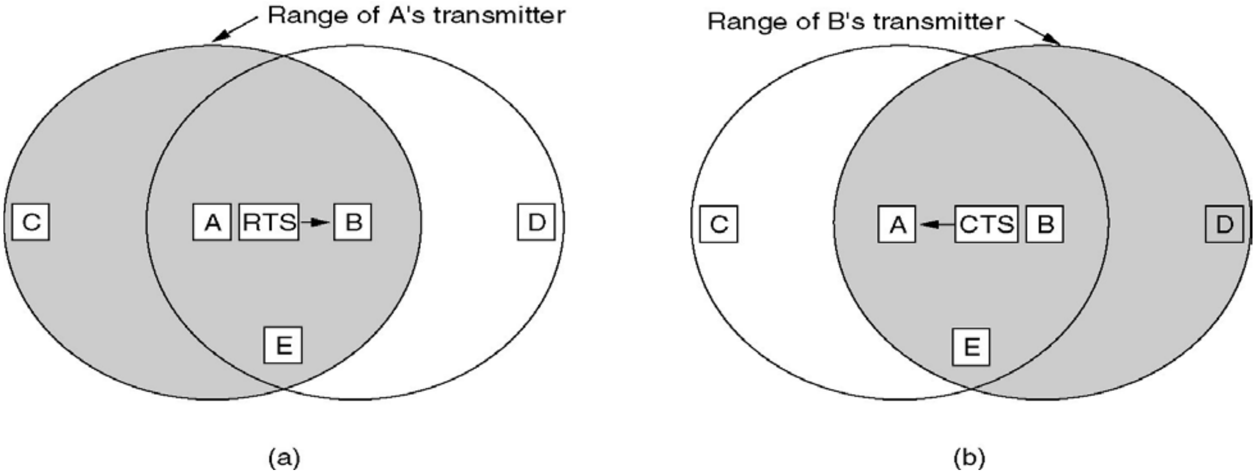
- a data channel for data packet transmissions
- a control channel used for transmitting the busy tone signal
- A node can transmit on the data channel only if it finds the busy tone to be absent on the control channel.The data packet is divided into two portions: a preamble and the actual data packet

Part – B

1. Explain MACAW protocol in detail. [CO2-H1]

- MACAW: A Media Access Protocol for Wireless LANs is based on MACA (Multiple Access Collision Avoidance) Protocol
- MACA
    - When a node wants to transmit a data packet, it first transmit a **RTS** (**Request To Send)** frame.
    - The receiver node, on receiving the RTS packet, if it is ready to receive the data packet, transmits a **CTS (Clear to Send)** packet.
    - Once the sender receives the CTS packet without any error, it starts transmitting the data packet.
    - If a packet transmitted by a node is lost, the node uses the binary exponential back-off (BEB) algorithm to back off a random interval of time before retrying.

- The binary exponential back-off mechanism used in MACA might starves flows sometimes. The problem is solved by MACAW.



The MACA protocol. (a) A sending an RTS to B.
(b) B responding with a CTS to A.

- Variants of this method can be found in IEEE 802.11 as DFWMAC (Distributed Foundation Wireless MAC),
- MACAW (MACA for Wireless) is a revision of MACA.
    - The sender senses the carrier to see and transmits a **RTS** (**Request To Send)** frame if no nearby station transmits a RTS.
    - The receiver replies with a **CTS** (**Clear To Send**) frame.
    - Neighbors
        - see CTS, then keep quiet.
        - see RTS but not CTS, then keep quiet until the CTS is back to the sender.
    - The receiver sends an ACK when receiving an frame.
        - Neighbors keep silent until see ACK.
    - Collisions
        - There is no collision detection.

- The senders know collision when they don’t receive CTS.
- They each wait for the exponential backoff time.

- A routing protocol for ad hoc wireless networks should have the following characteristics:
- It must be fully distributed as centralized routing involves high control overhead and hence is not scalable.
- It must be adaptive to frequent topology changes caused by the mobility of nodes.
- Route computation and maintenance must involve a minimum number of nodes. Each node in the network must have quick access to routes, that is, minimum connection setup time is desired
- It must be localized, as global state maintenance involves a huge state propagation control overhead
- It must be loop-free and free from state routes.
- The number of packet collisions must be kept to a minimum by limiting the number of broadcasts made by each node. The transmissions should be reliable to reduce message loss and to prevent the occurrence of state routes.
- It must converge to optimal routes once the network topology becomes stable. The convergence must be quick.
- It must optimally use scarce resources such as bandwidth, computing power, memory, and battery power.
- Every node in the network should try to store information regarding the stable local topology only.
- Changes in remote parts of the network must not cause updates in the topology information maintained by the node.
- It should be able to provide a certain level of quality of service (QoS) as demanded by the applications, and should also offer support for time-sensitive traffic.

2. Explain the contention based protocols with scheduling and reservation in detail. [CO2-H2]

Following are the main issues one should have in mind when considering designing a MAC protocol for ad hoc wireless networks. Bandwidth efficiency: The scarcity of bandwidth resources in these networks calls for its efficient usage. To quantify this, we could say that bandwidth efficiency is the ratio of the bandwidth utilized for data transmission to the total available bandwidth. In these terms, the target will be to maximize this value. Quality of service support: Providing QoS in these networks is very difficult, due to the high mobility of the nodes comprising them. Once a node moves out of another node’s reach, the reservation in it is lost. On the other hand, in these networks QoS is sometimes extremely important, for example in military environments. Therefore, QoS should be provided somehow, despite the characteristics of ad hoc networks. Synchronization: Some mechanism has to be found in order to provide synchronization among the nodes. Synchronization is important for regulating the bandwidth reservation. Hidden and exposed terminal problems: The reason for these two problems is the broadcast nature of the radio channel, namely, all the nodes within a node’s transmission range receive its transmission. Hidden terminal problem – two nodes that are outside each-other’s range perform simultaneous transmission to a node that is within the range of each of them, hence, there is a packet collision.

Hidden nodes mean increased probability of collision at a receiver, whereas exposed nodes may be denied channel access unnecessarily, which means underutilization of the bandwidth resources. Error-prone shared broadcast channel: In radio transmission, a node can listen to all traffic within its range. Therefore, when there is communication going on no other node should transmit, otherwise there would be interferences. Access to the physical medium should be granted only if there is no session going on. Nodes will often compete for the channel at the same time; therefore, there is high probability of collisions. The aim of a MAC protocol will be to minimize them, while maintaining fairness. No central coordination: in ad hoc networks, there is no central point of coordination due to the mobility of the nodes. Therefore, the control of the access to the channel must be distributed among them. In order for this to be coordinated, the nodes must exchange information. It is the responsibility of the MAC protocol to make sure this overhead is not a burden for the scarce bandwidth. Mobility of nodes: The mobility of the nodes is one of its key features. The QoS reservations or the exchanged information might become useless, due to node mobility. The MAC protocol must be such that mobility has as little influence as possible on the performance of the whole network. Signal propagation delay: Signal propagation delay is the amount of time needed for the transmission to reach the receiver. If the value of this parameter is considerable, a node may start transmitting, when in fact, transmission from other nodes is taking place, but it has not reached the node yet. The ad hoc networks that utilize synchronization, therefore, will have to expand the time slot to accommodate the propagation delay. Hardware constraints: Most radio-receivers are designed in such a way that only halfduplex communication can take place. When a node is transmitting, the power level of the outgoing signal is higher than any received signal; therefore, the node receives its own Page 3 transmission. Here, we can also add hardware switching time – time needed to shift from one mode to the other. 3. Design principles for a MAC protocol in ad hoc networks The operation of the protocol should be distributed through all the nodes. The protocol should provide QoS support for real-time traffic. The average delay for packet transmission should be as small as possible. The bandwidth should be utilized efficiently. Each node must have a fair share of the available bandwidth. Control overhead should be minimized. The hidden and exposed terminal problems should be minimized. The protocol must be scalable to large networks. Power control mechanisms are needed for efficient management of the energy consumption of the nodes. Adaptive data rate control should be provided – a node controls the rate of outgoing traffic in relation also to the network load and to the status of the other nodes. Directional antennas are encouraged, the advantages are reduced interference, increased spectrum reuse, and reduced power consumption. Time synchronization between the nodes should be provided. 4. Classification of MAC protocols for ad hoc networks Several criteria can be used for the classification of MAC protocols, such as time synchronization, initiation approach, and reservation approach. Ad hoc network protocols can be classified into three basic types [1]: - Contention-based protocols; - Contention-based protocols with reservation mechanisms; - Contention-based protocols with scheduling mechanisms; There are also some MAC protocols outside the above categories.

5. Contention-based protocols with reservation mechanisms Even though these protocols are contention-based, the contention takes place only during the bandwidth reservation phase. 5.1. Distributed Packet Reservation Multiple Access (D-PRMA) Protocol According to [1], D-PRMA is based on TDMA. The time division of the channel is done into frames, then further into slots, then further into minislots. Each minislot contains two control fields, RTS/BI – Request To Send / Busy Indication and CTS/BI – Request To Send / Busy Indication. Page 5 Figure 4: Time Division in the D-PRMA protocol The mechanism of competition for slots is such that a certain period at the beginning of every slot is reserved for carrier-sensing. The nodes compete for the first minislot in each slot. The winning one transmits a RTS packet through the RTS/BI part of the first minislot. The receiver responds by sending a CTS packet through the CTS/BI field. Thus, the node is granted all the subsequent minislots [1]. In addition to that, this very same slot in the subsequent frames is reserved for the same node, until it ends its transmission. Within a time slot, communication between the source and destination nodes is done either by Time Division Duplexing (TDD), or by Frequency Division Duplexing (FDD). There are two rules for the reservation, which prioritize voice traffic: - Contention for the first minislot is done with probability 1 for voice traffic, and a smaller probability for other traffic. - The reservation of a minislot brings reservation of the subsequent slots only if the winning node is a voice one. 5.2. Collision Avoidance Time Allocation (CATA) Protocol In this protocol, time is divided into frames, each frame into slots, and each slot into 5 minislots. The first four minislots are control ones, CMS, only the fifth is used for data transmission, DMS, and it is longer than the other ones. Figure 5: Time Division in the CATA protocol CATA supports broadcast, unicast, and multicast transmissions at the same time. CATA has two basic principles: - The receiver of a flow must inform other potential source nodes about the reservation of the slot, and also inform them about interferences in the slot. - Negative acknowledgements are used at the beginning of each slot for distributing slot reservation information to senders of broadcast or multicast sessions. The CMS1 and CMS2 are used to inform neighbors of

the receiving and the sending nodes accordingly about the reservation. The CMS3 and CMS4 are used for channel reservation [1]. Page 6 CATA provides support for collision-free broadcast and multicast traffic. 5.3. Hop Reservation Multiple Access (HRMA) Protocol HRMA is a time slot-reservation protocol where each slot is assigned a separate frequency channel. A handshake mechanism is used for reservation to enable node pairs to reserve a frequency hop, thus providing collision-free communication and avoiding the hidden terminal problem [1]. Figure 6: Time Division in the HRMA protocol One frequency channel is a dedicated synchronizing channel where nodes exchange information. The remaining frequency channels are paired, one channel in each pair is used for reservation and data packets, and the other one is used for acknowledgements. As mentioned above, each time slot has a frequency channel. The time slot is divided into four periods, each period is reserved for sending a particular kind of packet or its acknowledgement, depending on which frequency channel of the pair this time slot belongs to. After the handshaking is over, the two nodes communicate by sending data and ACKs on the very same frequency channels. When a new node wants to join the network, it listens to the dedicated frequency and gathers information. When a node wants to send data, it listens to the Hop Reservation (HR) period. If there is a packet there, it tries again after a random amount of time, otherwise it sends a RTS packet, and waits for the CTS acknowledgement packet in the CTS period of the corresponding frequency channel. 5.4. MACA with Piggy-backed Reservation (MACA/PR) Protocol There are three main components of this protocol: - A MAC protocol; - A reservation protocol; - A QoS routing protocol. MACA/PR differentiates between real-time packets and best-effort packets; it provides bandwidth to real-time traffic. Time is divided into slots that are asynchronous in nature and have different lengths. Each node records the transmit and receive reservations of its neighbors in a reservation table. A node that wants to transmit a non-real-time packet finds a free slot in the table. Then it waits for the same slot the next time around. If it is still free, it sends a RTS packet in the slot, expects a CTS packet, then sends the data and receives the acknowledgement still in the same slot. The RTS and CTS packets contain in them the amount of time that the data transmission is going to take place. In this way, the neighbors of the source and destination nodes can update their tables. Page 7 Figure 7: Packet exchange in MACA/PR For real-time traffic, the first part is identical until the first data packet is sent. Each data packet contains information about the reservation of the next data packet. This information is piggy-backed to it. Each acknowledging packet also contains this information. Thus, the neighbors of both communicating nodes can update the information. When the sender receives the acknowledgement, it makes sure that the reservation was successful. If several acknowledgements do not come, the sender assumes that the reservation has been lost and restarts the whole procedure. The acknowledgement refreshes the reservation; unrefreshed ones are simply dropped from the reservation table. The nodes exchange the information in their reservation tables; this eliminates the hidden terminal problem. This mechanism works as a TDM for real-time traffic, while best-effort packets are transmitted in the empty slots. When a new node joins the network, at first it learns about it by receiving the reservation tables from the neighbors. Then it behaves just like the others. Advantage: global synchronization not required. Drawback: the RTS-CTS-DATA-ACK exchange takes place in the same slot in different cycles; therefore, random empty slots are not utilized.

3. List and explain the issues in designing a MAC protocol for ad hoc wireless networks. [CO2-H2] .

Based on the routing information update mechanism

Ad hoc wireless network routing protocols can be classified into 3 major categories based on the routing information update mechanism. They are:

Proactive or table-driven routing protocols*:*

- Every node maintains the network topology information in the form of routing tables byperiodically exchanging routing information.
- Routing information is generally flooded in the whole network.
- Whenever a node requires a path to a destination, it runs an appropriate path-finding algorithm on the topology information it maintains.

Reactive or on-demand routing protocols**:**

- Do not maintain the network topology information.
- Obtain the necessary path when it is required, by using a connection establishment process.

Hybrid routing protocols:

- Combine the best features of the above two categories.
- Nodes within a certain distance from the node concerned, or within a particular geographical region, are said to be within the routing zone of the given node.
- For routing within this zone, a table-driven approach is used.
- For nodes that are located beyond this zone, an on-demand approach is used.

TABLE-DRIVEN ROUTING PROTOCOLS

- These protocols are extensions of the wired network routing protocols
- They maintain the global topology information in the form of tables at every node
- Tables are updated frequently in order to maintain consistent and accurate network state information
- Ex: Destination sequenced distance vector routing protocol (DSDV), wireless routing protocol (WRP), source-tree adaptive routing protocol (STAR) and cluster-head gateway switch routing protocol (CGSR).

Destination sequenced distance-vector routing protocol

- It is an enhanced version of the distributed Bellman -Ford algorithm where each node maintains a table that contains the shortest distance and the first node on the shortest path to every other node in the network.
- It incorporates table updates with increasing sequence number tags to prevent loops, to counter thecount-to-infinity problem, and for faster convergence.
- As it is a table-driven routing protocol, routes to all destinations are readily available at every node at all times.
- The tables are exchanged between neighbors at regular intervals to keep an up -to-date view of the network topology.
- The table updates are of two types:

Incremental updates: Takes a single network data packet unit (NDPU). These are used when a node does not observe significant changes in the local topology.
Full dumps: Takes multiple NDPUs. It is done either when the local topology changes significantly or when an incremental update requires more than a single NDPU.
□ Table updates are initiated by a destination with a new sequence number which is always greater than the previous one.

- Consider the example as shown in figure (a). Here node 1 is the source node and node 15 is the destination. As all the nodes maintain global topology information, the route is already available as shown in figure (b).
- Here the routing table node 1 indicates that the shortest route to the destination node is available through node 5 and the distance to it is 4 hops, as depicted in figure (b)
- The reconfiguration of a path used by an on-going data transfer session is handled by the protocol in the following way.
- The end node of the broken link initiates a table update message with the broken link's weight assigned to infinity (∞) and with a sequence number greater than the stored sequence number for that destination.
- Each node upon receiving an update with weight ∞, quickly disseminates it to its neighbors in order to propagate the broken-link information to the whole network.
- A node always assign an odd number to the link break update to differentiate it from the even sequence number generated by the destination.
- Figure 7.6 shows the case when node 11 moves from its current position.

.

Advantages

- Less delay involved in the route setup process.
- Mechanism of incremental update with sequence number tags makes the existing wired network protocols adaptable to ad hoc wireless networks.
- The updates are propagated throughout the network in order to maintain an up-to-date view of the network topology at all nodes.

Disadvantages

- The updates due to broken links lead to a heavy control overhead during high mobility.
- Even a small network with high mobility or a large network with low mobility can completely choke the available bandwidth.
- Suffers from excessive control overhead.

- In order to obtain information about a particular destination node, a node has to wait for a table update message initiated by the same destination node.
- This delay could result in state routing information at nodes.

Wireless Routing Protocol (WRP)

- WRP is similar to DSDV; it inherits the properties of the distributed bellman-ford algorithm.
- To counter the count-to-infinity problem and to enable faster convergence, it employs a unique method of maintaining information regarding the shortest distance to every destination node in the network and penultimate hop node on the path to every destination node.

- Maintains an up-to-date view of the network, every node has a readily available route to every destination node in the network.
- It differs from DSDV in table maintenance and in the update procedures.
- While DSDV maintains only one topology table, WRP uses a set of tables to maintain more accurate information.
- The table that are maintained by a node are :

Distance table (DT): contains the network view of the neighbors of a node. It contains a matrix where each element contains the distance and the penultimate node reported by the neighbor
for a particular destination.

Routing table (RT): contains the up-to-date view of the network for all known destinations. It keeps the shortest distance, the predecessor/penultimate node, the successor node, and a flag indicating the status of the path. The path status may be a simplest (correct) path or a loop (error), or destination node not marked (null).

Link cost table (LCT): contains the cost of relaying messages through each link. The cost of broken link is ∞.it also contains the number of update periods passed since the last successful update was received from that link.

Message retransmission list (MRL): contains an entry for every update message that is to be retransmitted and maintains a counter for each entry.

- After receiving the update message, a node not only updates the distance for transmitted neighbors but also checks the other neighbors' distance, hence convergence is much faster than DSDV.
- Consider the example shown in figure below, where the source of the route is node 1 and destination is node 15. As WRP proactively maintains the route to all destinations, the route to any destination node is readily available at the source node.
- From the routing table shown, the route from node 1 to node 15 has the next node as node 2. The predecessor node of 15 corresponding to this route is route 12.
- The predecessor information helps WRP to converge quickly during link breaks.
- When a node detects a link break, it sends an update message to its neighbors with the link cost of the broken link set to ∞. After receiving the update message; all affected nodes update their minimum distances to the corresponding nodes. The node that initiated the update message then finds an alternative route, if available from its DT. Figure 7.8 shows route maintenance in WRP.

Advantages

- WRP has the same advantages as that of DSDV.
- It has faster convergence and involves fewer table updates.

Disadvantages

- The complexity of maintenance of multiple tables demands a larger memory and greater processing power from nodes in the adhoc wireless network.
- It is not suitable for highly dynamic and also for very large ad hoc wireless networks.

Cluster-Head Gateway Switch Routing Protocol (CGSR)

- Uses a hierarchical network topology.
- CGSR organizes nodes into clusters, with coordination among the members of each cluster entrusted to a special node named *cluster-head.*
- This cluster-head is elected dynamically by employing a least cluster change (LCC) algorithm.
- According to this algorithm, a node ceases to be a cluster-head only if it comes under the range of another cluster-head, where the tie is broken either using the lowest ID or highest connectivity algorithm.
- Clustering provides a mechanism to allocate bandwidth, which is a limited resource, among different clusters, thereby improving reuse.
- A token-based scheduling is used within a cluster for sharing the bandwidth among the members of the cluster.
- CGRS assumes that all communication passes through the cluster-head. Communication between 2 clusters takes place through the common member nodes that are members of both the cluster are called *gateways.*
- A gateway is expected to be able to listen to multiple spreading codes that are currently in operation in the clusters in which the node exist as a member.
- A gateway conflict is said to occur when a cluster-head issues a token to a gateway over spreading codewhile the gateway is tuned to another code.
- Gateways that are capable of simultaneously communicating over two interfaces can avoid gateway conflicts.
- The performance of routing is influenced by token scheduling and code scheduling that is handled at cluster-heads and gateways, respectively.
- Every member node maintains a routing table containing the destination cluster-head for every node in the network.
- In addition to the cluster member table, each node maintains a routing table which keeps the list of next-hop nodes for reaching every destination cluster.
- The cluster routing protocol is used here.
- Figure below shows the cluster head, cluster gateways, and normal cluster member nodes in an ad hoc wireless network.

Advantages

- CGSR is a hierarchical routing scheme which enables partial coordination between nodes by electing cluster-heads.
- Better bandwidth utilization is possible.
- Easy to implement priority scheduling schemes with token scheduling and gateway code scheduling.

Disadvantages

- Increase in path length and instability in the system at high mobility when the rate of change of cluster-head is high.

- In order to avoid gateway conflicts, more resources are required.
- The power consumption at the cluster-head node is also a matter of concern.
- Lead to Frequent changes in the cluster-head, which may result in multiple path breaks.

4. List the important goals of designing a MAC protocol for ad hoc wireless networks. [CO2-H1]

- Reservation Time Division Multiple Access
  - every frame consists of N mini-slots and x data-slots
  - every station has its own mini-slot and can reserve up to k data-slots using this mini-slot (i.e. x = N * k).
  - other stations can send data in unused data-slots according to a round-robin sending scheme (best-effort traffic)
- Implicit reservation (PRMA - Packet Reservation Multiple Access):
  - a certain number of slots form a frame, frames are repeated
  - stations compete for empty slots according to the slotted aloha principle
  - once a station reserves a slot successfully, this slot is automatically assigned to this station in all following frames as long as the station has data to send
  - competition for this slots starts again as soon as the slot was empty in the last frame
- Collision avoidance time allocation protocol (CATA)
  - based on dynamic topology-dependent transmission scheduling
  - Nodes contend for and reserve time slots by means of a distributed reservation and handshake mechanism.
  - Support broadcast, unicast, and multicast transmissions.
  - The operation is based on two basic principles:
    - The receiver(s) of a flow must inform the potential source nodes about the reserved slot on which it is currently receiving packets. The source node must inform the potential destination node(s) about interferences in the slot.
    - Usage of negative acknowledgements for reservation requests, and control packet transmissions at the beginning of each slot, for distributing slot reservation information to senders of broadcast or multicast sessions.
- Hop reservation multiple access protocol (HRMA)
  - a multichannel MAC protocol which is based on half-duplex, very slow frequency-hopping spread spectrum (FHSS) radios
  - uses a reservation and handshake mechanism to enable a pair of communicating nodes to reserve a frequency hop, thereby guaranteeing collision-free data transmission.

    - can be viewed as a time slot reservation protocol where each time slot is assigned a separate frequency channel.
- Soft reservation multiple access with priority assignment (SRMA/PA)
    - Developed with the main objective of supporting integrated services of real-time and non-real-time application in ad hoc networks, at the same time maximizing the statistical multiplexing gain.
    - Nodes use a collision-avoidance handshake mechanism and a soft reservation mechanism.
- Five-Phase Reservation Protocol (FPRP)
    - a single-channel time division multiple access (TDMA)-based broadcast scheduling protocol.
    - Nodes uses a contention mechanism in order to acquire time slots.
    - The protocol assumes the availability of global time at all nodes.
    - The reservation takes five phases: reservation, collision report, reservation confirmation, reservation acknowledgement, and packing and elimination phase.
- MACA with Piggy-Backed Reservation (MACA/PR)
    - Provide real-time traffic support in multi-hop wireless networks
    - Based on the MACAW protocol with non-persistent CSMA
    - The main components of MACA/PR are:
        - A MAC protocol
        - A reservation protocol
        - A QoS routing protocol
- Real-Time Medium Access Control Protocol (RTMAC)
    - Provides a bandwidth reservation mechanism for supporting real-time traffic in ad hoc wireless networks
    - RTMAC has two components
        - A MAC layer protocol is a real-time extension of the IEEE 802.11 DCF.
            - A medium-access protocol for best-effort traffic
            - A reservation protocol for real-time traffic
        - A QoS routing protocol is responsible for end-to-end reservation and release of bandwidth resources.
- Distributed Wireless Ordering Protocol (DWOP)
    - A media access scheme along with a scheduling mechanism
    - Based on the distributed priority scheduling scheme
- Distributed Laxity-based Priority Scheduling (DLPS) Scheme
    - Scheduling decisions are made based on
    - The states of neighboring nodes and feed back from destination nodes regarding packet losses

  - Packets are recorded based on their uniform laxity budgets (ULBs) and the packet delivery ratios of the flows. The laxity of a packet is the time remaining before its deadline.
- MAC protocols that use directional antennas have several advantages:
  - Reduce signal interference
  - Increase in the system throughput
  - Improved channel reuse
- MAC protocol using directional antennas
  - Make use of an RTS/CTS exchange mechanism
  - Use directional antennas for transmitting and receiving data packets
- Directional Busy Tone-based MAC Protocol (DBTMA)
  - It uses directional antennas for transmitting the RTS, CTS, data frames, and the busy tones.
- Directional MAC Protocols for Ad Hoc Wireless Networks
  - DMAC-1, a directional antenna is used for transmitting RTS packets and omni-directional antenna for CTS packets.
  - DMAC-1, both directional RTS and omni-directional RTS transmission are used.
- Multi-channel MAC Protocol (MMAC)
  - Multiple channels for data transmission
  - There is no dedicated control channel.
  - Based on channel usage channels can be classified into three types: high preference channel (HIGH), medium preference channel (MID), low preference channel (LOW)
- Multi-channel CSMA MAC Protocol (MCSMA)
  - The available bandwidth is divided into several channels
- Power Control MAC Protocol (PCM) for Ad Hoc Networks
  - Allows nodes to vary their transmission power levels on a per-packet basis
- Receiver-based Autorate Protocol (RBAR)
  - Use a rate adaptation approach
- Interleaved Carrier-Sense Multiple Access Protocol (ICSMA)
  - The available bandwidth is split into tow equal channels
  - The handshaking process is interleaved between the two channels.

5. Illustrate various steps involved in five phase reservation protocol with its frame format. [CO2-H1]

Source-Tree Adaptive Routing Protocol (STAR)

□ Key concept-least overhead routing approach (LORA)

□ This protocol attempts to provide feasible paths that are not guaranteed to be optimal

□ Involves much less control overhead

□ In STAR protocol, every node broadcasts its source tree information

□ The source tree of a node consists of the wireless links used by the node.

□ Every node builds a partial graph of the topology.

□ During initialization, a node sends an update message to its neighbors

□ Each node will have a path to every destination node.

□ The path would be sub-optimal.

□ The data packet contains information about the path to be traversed in order to prevent the possibility of routing loop formation.

□ In the presence of a reliable broadcast mechanism, STAR assumes implicit route maintenance.

□ In addition to path breaks, the intermediate nodes are responsible for handling the routing loops

□ The RouteRepair packet contains the complete source tree of node k and the traversed path of the packet.

□ When an intermediate node receives a RouteRepair update message, it removes itself from the top of the route repair path and reliably sends it to the head of the route repair path.

Advantages

□ Very low communication overhead

□ Reduces the average control overhead

ON-DEMAND ROUTING PROTOCOLS

They execute the path-finding process and exchange routing information only when a path is required by a
node to communicate with a destination

Dynamic Source Routing Protocol (DSR)

Designed to restrict the bandwidth consumed by control packets in adhoc wireless networks by eliminating the periodic table update messages

□ It is beacon-less and does not require periodic hello packet transmissions

□ Basic approach to establish a route by flooding RouteRequest packets in the network.

□ Destination node responds by sending a Route Reply packet back to the source

□ Each Route Request carries a sequence number generated by the source node and the path it has traversed

□ A node checks the sequence number on the packet before forwarding it.

□ The packet is forwarded only if it is not a duplicate RouteRequest.

□ The sequence number on the packet is used to prevent loop formations and to avoid multiple transmissions.

□ Thus, all nodes except the destination forward a RouteRequest packet during the route construction phase.

□ In figure 7.10, source node 1 initiates a RouteRequest packet to obtain a path for destination node 15

□ This protocol uses a route cache that stores all possible information extracted from the source route contained in a data packet.

□ During network partitions, the affected nodes initiate RouteRequest packets.

□ DSR also allows piggy-backing of a data packet on the RouteRequest.

□ As a part of optimizations, if the intermediate nodes are also allowed to originate RouteReply packets,

then a source node may receive multiple replies from intermediate nodes.
In fig 7.11, if the intermediate node 10 has a route to the destination via node 14, it also sends the RouteReply to the source node.

□ The source node selects the latest and best route and uses that for sending data packets.

□ Each data packet carries the complete path to its destination.

□ If a link breaks, source node again initiates the route discovery process

Advantages

□ Uses a reactive approach which eliminates the need to periodically flood the network with table update messages.

□ Route is established only when required.

□ Reduce control overhead

Disadvantages

□ Route maintenance mechanism does not locally repair a broken link

□ Stale route cache information could result in inconsistencies during route construction phase

□ Connection set up delay is higher

□ Performance degrades rapidly with increasing mobility

□ Routing overhead is more & directly proportional to path length

Ad Hoc On-Demand Distance Vector Routing Protocol

□ Route is established only when it is required by a source node for transmitting data packets

□ It employs destination sequence numbers to identify the most recent path

□ Source node and intermediate nodes store the next hop information corresponding to each flow for data packet transmission

□ Uses DestSeqNum to determine an up-to-date path to the destination.

□ A RouteRequest carries the source identifier, the destination identifier, the source sequence number, the destination sequence number, the broadcast identifier and the time to live field.

□ DestSeqNum indicates the freshness of the route that is accepted by the source.

□ When an intermediate node receives a RouteRequest, it either forwards it or prepares a RouteReply if it has a valid route to the destination.

□ The validity of the intermediate node is determined by comparing the sequence numbers.

□ If a RouteRequest is received multiple times, then duplicate copies are discarded.

□ Every intermediate node enters the previous node address and its BcastID.

□ A timer is used to delete this entry in case a RouteReply packet is not received.

□ AODV does not repair a broken path locally

□ When a link breaks, the end nodes are notified

□ Source node re-establishes the route to the destination if required

Advantage

□ Routes are established on demand and DestSeqNum are used to find latest route to the destination

□ Connection setup delay is less

Disadvantages

□ Intermediate nodes can lead to inconsistent routes if the source sequence number is very old.

□ Multiple RouteReply packets to single RouteRequest packet can lead to heavy control overhead.

□ Periodic beaconing leads to unnecessary bandwidth consumption.

Temporally Ordered Routing Algorithm (TORA)

□ Source-initiated on-demand routing protocol.

□ Uses a link reversal algorithm.

□Provides loop free multi path routes to the destination.

□ Each node maintains its one-loop local topology information.

□ Has capability to detect partitions.

□ Unique property limiting the control packets to a small region during the reconfiguration process initiated by a path break

6. How is scheduling mechanism achieved in distributed wireless ordering protocol? Explain in detail. How are Information symmetry and perceived collisions handled? [CO2-H2]

MACAW PROTOCOL • The sender senses the carrier to see and transmits a RTS (Request To Send) frame if no nearby station transmits a RTS. • The receiver replies with a CTS (Clear To Send) frame. • The MACAW protocol uses one more control packet called the request-for-request-to-send (RRTS)

Neighbors • see CTS, then keep quiet. • see RTS but not CTS, then keep quiet until the CTS is back to the sender. • The receiver sends an ACK when receiving an frame. • Neighbors keep silent until see ACK. • Collisions • There is no collision detection. • The senders know collision when they don't receive CTS. • They each wait for the exponential back-off time. MACA-By Invitation Protocol It is a receiver-initiated protocol¬ It reduces the number of control packets used in the MACA protocol¬ It eliminated the need for the RTS packet¬ In MACA-BI, the receiver node initiates data transmission by transmitting a ready-to-receive¬ (RTR) control packet to the sender as shown in the figure

If it is ready to transmit, the sender node respond by sending a DATA packet • Thus data transmission in MACA-BI occurs through a two-way handshake mechanism¬ The efficiency of the MACA-BI scheme is mainly dependent on the ability of the receiver node to¬ predict accurately the arrival rates of traffic at the sender nodes

A common problem observed in wireless multihop networks is a situation where externally offered load entering the network exceeds the network capacity. If the network capacity is exceeded, packets are queued en-route to the receiver resulting in higher end-to-end packet delays, and wastage of bandwidth when packets are dropped at intermediate nodes. Unfair distribution of bandwidth among users is another challenge that a network designer needs to address specially in distributed ad-hoc and mesh networks. In this context, an appropriate and viable solution is a maxmin fair rate allocation[5] in which resources are allocated in order of increasing demand such that no user gets a resource share larger than its demand and users with unsatisfied demands get an equal share of the resource. Also a user with

unsatisfied demands cannot increase its resource share without reducing the share of others who are already using equal or lesser amount of the resource. Our goal in this paper is to develop a distributed max-min fair queuing mechanism that enforces this notion of fairness for multihop flows in wireless mesh networks. We compute the maxmin fair rate of a multihop flow by computing the maxmin fair rate at each hop along its path and finally enforcing the rate offered to the flow at the most constrained hop in the path. This approach provides the framework for a multihop maxmin fair rate allocation as well as bounds the rate at which packets are injected in the network to the maximum rate at which it can be delivered to the destination. Although our queuing mechanism can work with any reasonable MAC protocol, we find that the IEEE 802.11 MAC seriously deviates from fairness principles in certain scenarios [4],[11],[2]. In order to reduce MAC layer unfairness, we replace the exponential backoff mechanism in 802.11, with virtual time based CSMA (VTCSMA) which is a backoff scheme based upon packet arrival time. VTCSMA [7] provides a distributed first come first serve medium access to contending nodes. This approach ensures that the scheduling order computed at the upper layer is also enforced in the MAC layer. The VTCSMA protocol was designed for single hop networks, and our work extends it for multihop networks. This is nontrivial as problems such as hidden terminals and starvation must be addressed. Our queuing method and the MAC layer protocol together form a complete protocol suite that computes and enforces max-min fair scheduling in wireless mesh networks in a distributed manner. The rest of the paper is organized as follows. In section 2, we will explain the background, theory and definition of max-min flow control in the context of wireless multihop networks. We will then describe our upper layer protocol in section 3 followed by the MAC layer solution in section 4. We present performance evaluation in section 5 and related work and conclusions in sections 6 and 7.In wireless networks, transmission between a pair of neighboring nodes (also called single hop flow) interferes with a transmission between another pair if either the two single hop flows have a common transmitter or receiver or if the transmitter or receiver of one is within two hop distance from the transmitter or receiver of the other. The two hop consideration is due to the assumption of an 802.11- like protocol where any transmission can interfere up to two hops. We model these interfering flows using a contention graph, henceforth called flow contention graph, where nodes are single hop flows on the network graph and edges are drawn between two nodes if the flows interfere. An example of the flow contention graph is shown in Figure 1. Given this notion of flow contention graph, earlier work [2] has considered max-min fair rate of single hop flows. In our work, we consider end-to-end multihop flows as multiple single hop flows that can go over a sequence of links. We first treat these single hop segments as individual flows and then extend the idea of fairness to multihop flows. To demonstrate the technique let us first describe the notion of feasibility and max-min fair allocation. A feasible rate allocation essentially constrains the rate allocation for each flow such that the sum total of the rates allocated to all flows belonging to a clique in the flow contention graph do not exceed the network capacity. A rate allocation is max-min fair if it is feasible and the only way a flow can get higher rate is by reducing the rate of some other flow that has been allocated equal or lower rate. Formal definitions are below. Definition 1 (Feasible Rate Vector) Assume that C is the link capacity in the wireless network. Let R represent a vector that represents transmission rates ri allocated to each flow fi in a “clique” in the flow contention graph. If F is the set of flows in the clique, then the vector R of rates ri is feasible if ri ≥ 0, X ∀fi ri ≤ C. Definition 2 (Max-min Fair Rate Allocation) A feasible rate vector is max-min fair if for any flow fi , the allocated rate ri cannot be increased while maintaining feasibility without decreasing rj for some flow fj for which rj ≤ ri [1]. Flows fi and fj do not need to belong to the same clique. Prior work [2] has shown that a feasible rate vector R is max-min fair if and only if each flow has a bottleneck clique with respect to R. Bottleneck clique is defined as follows. Definition 3 (Bottleneck Clique) Given a max-min fair rate vector R, a bottleneck clique cli is that clique for which flow fi ∈ cli , P ∀fk∈cli rk = C, and allocated rate ri of fi is equal or greater than the allocated rate rk of any other fk ∈ cli . The largest clique in the network is the bottleneck clique for the flows it contains. 2.1 Max-Min Rate Calculation Based on the above, prior work [2] has provided a mechanism to compute max-min fair allocation of rates on single hop flows in the network. The technique simply determines all cliques in the flow contention graph. Since this can be computationally intractable, heuristics are used for the clique computation. Starting with the largest clique, each flow in the clique is allocated equal share of the remaining capacity except the ones that have already received an allocation. The remaining capacity is simply the capacity C minus the already allocated rates. The allocation is started with the largest clique, as this clique is always the bottleneck for the flows belonging to this clique and thus determines the fair rate allocation of these flows. For the benefit of the reader, we illustrate the procedure using the example of Figure 2. Assume capacity C = 1. There are three cliques with 3, 4 and 7 nodes respectively with some common vertices’s (A, B, C) corresponding to network flows. The procedure starts with clique 3, assigning a rate of 1 7 to each vertex of clique 3. Then it turns to clique 2. Since B and C have already been allocated their rates, A and D are allocated the remaining capacity equally. Each of them gets 1 2 (1 − 2 7) = 5 14 . But since the rate allocated to A by clique 1 is only 1 3 which is less than the rate being offered by clique 2, it receives only 1 3 rate, while node D finally gets 1 − 2 × 1 7 − 1 3 = 8 21 part of the bandwidth. 3 Upper layer Protocol to achieve Max-min fair scheduling In the prior section, we have described how to compute max-min fair rates for single hop flows in the network. In this section, we develop a queuing mechanism that computes and allocates max-min fair rates to multihop flows. The protocol has three components: “clique formation protocol” that computes the allocations locally on single hop segments of multihop flows; “back pressure protocol” that assigns fair rates to multihop flows; “rate enforcement protocol” which essentially controls the scheduling and enforces that no flow exceeds its allocated rate. 3.1 Clique Formation Protocol In order to compute fair rates for all flows in the network in a distributed fashion, each network node needs to obtain the flow contention graph that represents its local neighborhood. The local neighborhood of a node consists of its neighbors that can be reached in up to two hops. A two-hop message exchange protocol gathers enough information to build the local flow contention graph. This can be done by sending “hello” messages and rebroadcasting the contents so that the two-hop neighbors of the original sender can receive the messages as well. These “hello” messages are similar to “hello” messages that many routing protocols (e.g., AODV [8]) employ to maintain neighborhood information; so we do not consider them to be additional overheads except the additional content. Frequency of such exchange for our protocol objective should be the granularity of any topology change or traffic changes (in terms of origination of a new flow or expiry of an existing flow). Each node i maintains and includes in the “hello” messages, information about the single hop flows that a node originates, receives or routes. These single hop flows may be segments of multihop flows. This information includes the flow id (fm), the nexthop receiver of the flow (node j) and the rate allocated to the flow (rm,i) at node i. Thus, the “hello” messages contain a set of tuples fm,i,j =< fm, j, rm,i >. We will refer to the set of fm,i,j tuples as the local flow set (Li) for node i. Apart from Li , node i also includes in the “hello” messages, the same information about the flows that interfere with its transmissions. We will refer to this set as the interfering flow set or (Ii). The Ii is the union of local flow sets Lj of all nodes within the two hop neighborhood of node i. Thus, if Ni is the set of one and two hop neighbors of node i then, Ii = [∀j∈Ni Lj . 1 (1) After receiving messages from all neighbors, node i is able to construct a neighbors interfering set or Pi such that, Pi = [∀j∈Ni Ij . (2) This information is sufficient [2] for node i to compute the flow contention graph representing its neighborhood and calculate all cliques in this graph. The fair share of bandwidth of all members of the bottleneck clique in the network is simply the ratio of the bandwidth and the size of the clique [2]. We cannot obtain the size or content of the bottleneck clique in the entire network due to the hardness of the problem. But we can find all cliques and compute the bottleneck clique in the local neighborhood consisting of few nodes, in reasonable time. Thus, for every flow the node keeps track of the local bottleneck clique corresponding to that flow and computes rate, say S. If after subsequent “hello” message exchanges, the node sees that other flows in this clique insist on getting less than rate S, it redistributes the residual rate among other flows in the clique and recomputes the local bottleneck clique. Thus, we may claim that, at the steady state, the rate of each flow in the network is equal to that offered by the flow’s local bottleneck clique which is the max-min fair rate of the flow. Let us explain this with an example in Figure 2. This figure represents a flow contention graph of the network. Clique 3 is the largest clique in the network and thus is a local bottleneck clique for all member flows. Flow A in the graph is a member of both clique 1 and clique 2. The rates offered by the cliques to flow A are 1 3 and 5 14 respectively. Thus although clique 2 is the largest clique for flow A in terms of size, clique 1 is the bottleneck clique as it allows a rate lower than clique 2. 3.2 Back Pressure Protocol In the previous section, we treated multi-hop flows as multiple single hop segments of the flow thereby assigning rates to each segment of the flow at the local bottleneck cliques. We now introduce the notion of a global bottleneck 1Here we would like to mention that when computing the union or intersect of sets, a node only considers the < fm, j > pair from the tuple while rm,i is used in rate computations at upstream and downstream clique for multihop flows as the clique at which the flow receives the least rate along its path. A more formal definition is as follows. Definition 4 (Global Bottleneck Clique) A global bottleneck clique for a multihop flow is the clique containing the single hop flow segment fm,i,j (flow id m, from node i to node j) of the multihop flow Fm,a,b (flow id m, from source a to destination b), where the offered rate Sm,i,j at node i is less than the rate offered at any other node k along the flow's path. Consider Figure 2 again. A multihop flow F in the figure is represented by three single hop flow segments – f1, f2 and f3. The rate offered at each of these segments are 1 3 , 1 3 and 1 7 respectively. Thus clique 3 is the global bottleneck clique for flow F since it offers the least rate compared to other cliques along the path from source to destination. If the rate provided at upstream nodes of a multihop flow is larger than the rate offered at the global bottleneck clique, packets may be queued and dropped at the forwarding nodes. Similarly, if the rate offered at downstream nodes is higher than the rate allocated at the global bottleneck clique, the allocated rate will remain unused instead of being utilized by other flows with unfulfilled demands. In order to prevent such wastage of bandwidth, we introduce a back pressure protocol in which each node limits a multihop flow's rate to the minimum of the rates provided at the next hop, at the previous hop and at the current hop. The source and destination of the multihop flow, limit the flow's rate to the minimum of the computed rate and that offered at the next or previous hop respectively. This scheme achieves what the authors in the paper [9] have tried to achieve by a more complex token generation process. Due to this back pressure mechanism, the rate offered by the global bottleneck clique for the flow is propagated to all nodes along the path from the source to the destination of the flow. The extra bandwidth available after applying the back pressure technique is distributed among other flows after the next hello message exchange and the local and global bottleneck cliques are recomputed. A detailed mathematical analysis of the token based back pressure technique is presented in [9] which also applies to our technique. 3.3 Rate Enforcement Protocol In order to enforce the assigned rates, the protocol needs to ensure that the rate at which the packets are transmitted follows the rate computed by the clique formation protocol and the back pressure protocol. We employ a timer based mechanism to "release" packets at the computed rate. A flow may be served only if there is a packet that has been "released" for transmission. Every node that has packets to send, runs a timer, which we will refer to as the release timer. The interval of release timer is calculated dynamically and depends upon the number of contending flows in the local neighborhood. When the release timer fires, the node checks if there is a flow from which a packet can be “released”. A packet can be “released” if the flow to which the packet belongs has used less than its allocated rate otherwise the next flow is considered. This scheme ensures that each flow receives no more that the rate computed by the clique formation and back pressure protocols, thereby enforcing the computed rates. 4 Virtual Time Based MAC Protocol The three step upper layer protocol that we proposed in the previous section can be used in conjunction with any reasonable MAC layer protocol in wireless network. However, we know from [11],[4],[2] that the commonly used IEEE 802.11 MAC protocol suffers from several unfairness issues. This is due to several reasons including exposed terminals, hidden terminals and the backoff policy used in 802.11. We have developed a medium access protocol to complement our scheduling scheme. Our MAC protocol performs a packet arrival based backoff mechanism known as virtual time CSMA (VTCSMA) [7] rather than random exponential backoff mechanism used in 802.11. The VTCSMA MAC protocol implements a first come, first serve access to the shared medium by emulating a single server multiple queue system. Only here the queues are maintained at different nodes in the network and the scheduling decision must be made in a distributed manner. In order to achieve this distributed scheduling process, each node in the network maintains two clocks, real clock and virtual clock, to measure the passage of real time and virtual time respectively. Both clocks may be initialized to zero and the real clock runs at a constant rate. The virtual clock runs η times faster than the real time clock while the medium is idle (unless the two clocks are in sync, in which case they run in lock steps). The virtual clock is stopped whenever the medium becomes busy and it resumes when the medium is idle again. When the virtual clock of a node passes the arrival time of the packet in the head of its queue, the packet is transmitted. If all nodes in the network share the same wireless medium and follow this transmission rule, the first-come first-serve scheduling is trivially achieved in a distributed manner. The analysis in [7] shows that this protocol can potentially provide a higher goodput as compared to random access CSMA. VTCSMA as described above provides fair medium access when all nodes are within a single collision domain i.e., all nodes are within receive range of one another. Since in a single collision domain, nodes can “hear” transmissions from each other, the virtual clocks run almost in sync or atleast at the same average rate. The average

7. What are the advantages of reservation based MAC protocol over contention based MAC Protocol? [CO2-H1]

Virtual Time Based MAC Protocol The three step upper layer protocol that we proposed in the previous section can be used in conjunction with any reasonable MAC layer protocol in wireless network. However, we know from [11],[4],[2] that the commonly used IEEE 802.11 MAC protocol suffers from several unfairness issues. This is due to several reasons including exposed terminals, hidden terminals and the backoff policy used in 802.11. We have developed a medium access protocol to complement our scheduling scheme. Our MAC protocol performs a packet arrival based backoff mechanism known as virtual time CSMA (VTCSMA) [7] rather than random exponential backoff mechanism used in 802.11. The VTCSMA MAC protocol implements a first come, first serve access to the shared medium by emulating a single server multiple queue system. Only here the queues are maintained at different nodes in the network and the scheduling decision must be made in a distributed manner. In order to achieve this distributed scheduling process, each node in the network maintains two clocks, real clock and virtual clock, to measure the passage of real time and virtual time respectively. Both clocks may be initialized to zero and the real clock runs at a constant rate. The virtual clock runs η times faster than the real time clock while the medium is idle (unless the two clocks are in sync, in which case they run in lock steps). The virtual clock is stopped whenever the medium becomes busy and it resumes when the medium is idle again. When the virtual clock of a node passes the arrival time of the packet in the head of its queue, the packet is transmitted. If all nodes in the network share the same wireless medium and follow this transmission rule, the first-come first-serve scheduling is trivially achieved in a distributed manner. The analysis in [7] shows that this protocol can potentially provide a higher goodput as compared to random access CSMA. VTCSMA as described above provides fair medium access when all nodes are within a single collision domain i.e., all nodes are within receive range of one another. Since in a single collision domain, nodes can “hear” transmissions from each other, the virtual clocks run almost in sync or atleast at the same average rate. The average rate is calcu lated as the rate at which the virtual time progresses with respect to progress of real time. The average rate of virtual clock at any node depends upon the contention level it experiences. Also since a packet is transmitted only when the virtual time reaches the packet arrival time, the throughput achieved by a node is also a function of the average rate of the virtual clock. In a multihop network, the contention experienced by nodes differ from one region to another. It is easy to construct scenarios where some nodes experience larger contention than their neighbors thereby getting fewer chances to transmit than other nodes. This phenomenon may lead to unfair share of bandwidth and even starvation. Figure 3(c) shows a typical scenario where this may happen. Here node 5 being in the carrier sensing range of both nodes 0 and 3, faces higher contention than either node 0 or node 3 which do not contend with one another.

Therefore, the average rate of node 5’s virtual clock is lower than that of 0 and 3. We suggest a two step approach to address this problem in the multihop extension of the VTCSMA protocol described in the next section. 4.1 VTCSMA in Wireless Multihop Networks We have proposed a multihop VTCSMA MAC protocol that alleviates the starvation problem of VTCSMA. We borrow the virtual carrier sensing and solution to hidden terminal problem from IEEE 802.11 where nodes maintain “network allocation vectors (NAV)” and exchange RTS/CTS control packets to maintain channel state and to notify potential interferers of the impending transmission. To solve the starvation problem in VTCSMA, we propose that every packet must carry the virtual time stamp of the transmitting node and every node in the network must follow a two step approach to prevent starvation. In the first step which we name “good neighbor approach”, nodes reduce the possibility of starvation of their neighbors by adjusting their virtual clock to minimum of the virtual time stamp from overheard packets and the time measured by the local virtual clock. The second step which we name “bad neighbor approach” is invoked when a node that has packets to transmit, overhears another packet with a virtual time stamp that is ahead of its own virtual time by more than a fixed threshold (an indication of starvation). The starving node then sends a jamming message that conveys this situation to all receivers in its vicinity, forcing all nodes to invoke their collision recovery mechanism i.e setting the NAV and withholding all transmissions. Here we propose an additional network allocation vector called “soft NAV”. When a node detects a jamming signal or a collision, it waits for the medium to become idle again and then sets a “soft NAV” in addition to the regular NAV. During this “soft NAV” state or “soft state”, nodes do not run their virtual clock and do not initiate any transmission, but they may receive unicast transmissions and send acknowledgements. While neighboring nodes are in the ”soft state”, the starving node gets the opportunity to transmit its backlogged packets. At this time, nodes with faster virtual clocks adjust their clocks in the manner of the “good neighbor approach”. This two step approach is instrumental in reducing the difference between average rate of virtual clocks in the network which prevents starvation in the network. 5 Results We evaluated the performance of our queuing protocol and compared with a first-come-first-serve scheduling mechanism that schedules packets in the order they arrive in the queue at each node without consideration for the flow to which they belong. We have also compared the performance of the two MAC protocols in conjunction with each scheduling protocol. We used fairness index and goodput as the metrics to evaluate performance. Definition 5 (Fairness Index) If a system allocates resources to n contending users, such that the i th user receives an allocation xi , then fairness index is defined as f(x) = (Pn i=1 xi) 2 n Pn i=1 x 2 i , xi ≥ 0. Definition 6 (Goodput) Goodput is defined as the number of application layer data bits successfully received at the receiver over the total span of time for which the application layer sent data. We have used network simulator ns2 version 2.27 [3] for all simulations. We have experimented with both small scenarios that represent specific problems that arise in multihop networks as well as random scenarios with varying packet rates and number of traffic sources. 5.1 Max-min Fair vs FCFS Scheduling with IEEE 802.11 We placed 7 nodes in a network as shown in Figure 3(a). We set up two TCP flows in the network, flow 1 from node 0 to node 6 and flow 2 from node 3 to node 6. We present the result of this experiment in table 1. We observe that the max-min fair scheduling protocol distributes the bandwidth more evenly between the two flows with flow 1 achieving a rate of 53kbps and flow 2 achieving 51kbps, but in FCFS scheduling, flow 1 receives a goodput of 169kbps while flow 2 is starved. In the network shown in Figure 3(b) two UDP flows represent the information asymmetry (IA) scenario [4]. Here, node 1 that originates flow 1 is within the carrier sensing range of node 4 which receives flow 2. On the other hand, node 3 that originates flow 2 does not have any information about flow 1 because it is beyond the transmission range of both node 1 and node 0. Since node 3 is unaware of transmissions by node 1, it is possible that node 3 attempts to transmit data while a transmission between nodes 1 and 0 is going on. These transmissions from node 3 may not be received correctly at node 4 due to interference with transmissions from node 1 causing multiple retransmission attempts by node 3. These retransmissions, in 802.11 based MAC protocols, lead to a larger contention window at the sender thus reducing its probability of acquiring the medium. This is reflected in the results shown in Figure 4(a), where the goodput achieved by flow 1 is more than 75% larger than that achieved by flow 2. In Figure 3(c),we constructed a perceived collision [4] scenario with UDP flows from node 0 to node 1, node 3 to node 4 and node 5 to node 6. In a perceived collision sceTable 1. Goodput vs load for symmetric scenario of Figure 3(a) with two TCP flows from node 0 to node 6 and node 3 to node 6 Flow FCFS Queue(Kbps) Fair Queue(Kbps) 1 169.46579 52.94678 2 0.70691 51.1774 nario, three flows ‘1’, ‘2’ and ‘3’ are such that flows ‘1’ and ‘2’ do not contend with one another but flow ‘3’, contends with both flows ‘1’ and ‘2’. Since the flow in the middle has to defer for the flows on each side, and therefore faces more contention compared to the neighboring flows, it gets fewer chances to transmit packets. Results in Figure 4(b) show that the middle flow receives very little share of the bandwidth while flows ‘1’ and ‘2’ each are able to receive 80% higher bandwidth share. When maxmin fair scheduling is used in both information asymmetry and perceived collision scenarios, we observe that the contending flows form a clique in the network and thus equally divide the bandwidth among each other thereby achieving nearly equal goodputs as shown in Figure 4(a) and Figure 4(b). 5.1.1 Multihop VTCSMA vs IEEE 802.11 We performed some experiments to demonstrate the advantage of using multihop VTCSMA over IEEE 802.11. We randomly placed 50 nodes in a network of size 1500x300m. Each node in the network transmits packets to a randomly selected neighbor. The virtual clock rate in VTCSMA is 200 times the real clock rate. The packet size is 512 bytes and we vary packet rates and compare fairness index and goodput for multihop VTCSMA and IEEE 802.11 in Figure 5(b) and Figure

5(a) respectively. We observe that VTCSMA achieves nearly perfect fairness index but lower goodput compared to 802.11. Here 802.11 achieves a higher goodput compared to VTCSMA but the fairness index graph shows that this is at the cost of unfair distribution of bandwidth among flows. The lower bandwidth utilization in fair scheduling protocols is due to the conflicting nature of the two goals. In [6] the author explains the difficulty of simultaneously achieving both fairness and maximizing bandwidth usage. 5.1.2 Maxmin and FCFS Scheduling with Multihop VTCSMA and IEEE 802.11 We randomly placed 50 nodes in a network of size 1500x300m and selected multihop flows between random pairs of nodes in the network. We experimented with 5,10,15 and 20 traffic connections that transmit UDP packets of size 1000 bytes at a rate of 10pkts/s. We compared the goodputs and fairness index5 of the two scheduling protocols under varying load conditions and the plots are shown in Figure 6(a) and Figure 6(b). We observe that with 20 traffic sources, maxmin scheduling with VTCSMA MAC provides a fairness index above 0.9 while fairness index in maxmin scheduling with 802.11 MAC protocol drops to 0.8. FCFS with VTCSMA is more fair compared to FCFS with 802.11. Also note that max-min fair scheduling with VTCSMA in the MAC layer outperforms all combinations in terms of both fairness index and goodput. These results clearly demonstrate the advantages of the protocol suite that we have proposed in this work. 6 Related Work Fair scheduling of flows in a wireless multihop network has been a popular topic of research for several years. In some of the earlier works, researchers have focused on providing a MAC layer solution for fair bandwidth allocation. In [11] the authors have proposed a scheduling discipline to schedule packets on an arrival time and packet size basis with concepts similar to virtual time CSMA. We discussed earlier in this paper the drawbacks of using virtual time for scheduling in multihop networks. Since this scheme was suggested for wireless LAN, the authors did not discuss the problems that may arise in wireless multihop networks. Similarly the scheme suggested in [10] and [4] schedules packets on a priority order, where the priorities are learned from information piggy backed on control and data packets. These papers also provide MAC layer solutions and fairness is achieved by appropriate backoff policy. In [6], the authors have provided a two tier solution to provide maxmin fair allocation for local flows and to maximize the network throughput. In the first step, the protocol achieves the fairness model by selecting a set of flows and then in the second step, the protocol tries to maximize the bandwidth utilization by scheduling the maximum independent set subject to the selection of the flows in the first phase. Since the problem of finding the maximum independent set is NP-complete, the authors implement a minimum degree greedy algorithm. The distributed implementation of the global model proposed in the paper requires that each time there is a change, the new information must be disseminated throughout the network in order to maximize network throughput. A backoff based protocol is used to achieve the local fairness model and to implement the minimum degree greedy algorithm for maximizing bandwidth utilization. In [2] the authors allocate maxmin fair rate to single hop flows in a multihop network and the fair rate of each flow is limited by the share provided by the bottleneck clique. The fair rate of a flow is calculated by computing the rate provided by the largest clique in the flow's flow contention graph and the fair rates are achieved by a backoff based MAC protocol. The authors in [9] present an algorithmic perspective of max-min fair allocation in wireless multihop networks. The network model used in this work is different from what we used in our work. Here each node in the network has a locally unique frequency, thus there is no location dependent contention. Unlike [2], flows are multihop flows and the fair rate of a flow in the network is limited by the share provided by the bottleneck link along the path of the flow.

- RTS, CTS and ACK control packets are used for transmitting best effort packets □.
- Time is divided into superframes. (figure)
- Bandwidth reservations can be made by a node by reserving variable-length time slots on superframes.
- The core concept of RTMAC is the flexibility of slot placement in the superframe.
- Each superframe consists of a number of reservation-slots.
- The time duration of each resv-slot is twice the maximum propagation delay.
- Data transmission normally requires a block of resv-slots.
- A node that needs to transmit real-time packets first reserves a set of resv-slots.The set of resv-slots reserved by a node for a connection on a superframe is called a connection-slot.
- Each node maintains a reservation table containing information such as the sender id, receiver id, and starting and ending times of reservations that are currently active
- In RTMAC, no time synchronization is assumed
- The protocol uses relative time for all reservation purpose
- A three way handshake protocol is used for effecting the reservation.
- In the figure, NAV indicates the network allocation vector maintained at each node.
- Main advantage is Bandwidth efficiency.
- Another advantage is asynchronous mode of operation where nodes do not require any global time synchronization.

Unit -III

ROUTING PROTOCOLS AND TRANSPORT LAYER IN AD HOC WIRELESS NETWORKS

Part – A

1.What are the responsibilities of routing protocol? [CO3-L1]

A **routing protocol** shares this information first among immediate neighbors, and then throughout the network. ... Interior gateway **protocols** type 1, link-state **routing protocols**, such as OSPF and IS-IS. Interior gateway **protocols** type 2, distance-vector **routing protocols**, such as **Routing** Information **Protocol**, RIPv2, IGRP

2. What are the major challenges in designing routing protocols? [CO3-L2]

The hidden terminal problem refers to the collision of packets at a receiving node due to simultaneous transmission of those nodes that are not within the direct transmission range of the sender, but are within the transmission range of receiver. Collision occurs when both nodes transmit packets at the same time [6]. The hidden and exposed terminal problems significantly reduce the throughput of a network when the traffic load is high. It is therefore desirable that the MAC protocol be free from the hidden and exposed terminal problems.

3. Differentiate proactive and reactive protocols. Write examples for each. [CO3-L2]

• Average end-to-end delay or the time taken by the data to reach the destination from the source is variable in Reactive Protocols but remains constant in Proactive Protocols for a given Ad hoc network.

• The delivery of packet data is much more efficient in Reactive Protocols than in Proactive Protocols.

• Reactive Protocols are much faster in performance than Proactive protocols.

• Reactive Protocols are much more adaptive and work much better in different topographies than Proactive Protocols.

4. List the characteristics of a routing protocol for ad hoc wireless networks. [CO3-L1]

Dynamic topologies The topic refers to the most essential property of an ad hoc network: Nodes can move arbitrarily with respect to other nodes in the network. 2. Bandwidth-constrained Nodes in an ad hoc network are mobile. Thus, they are using radio links that have far lower capacity than hardwired links could use. In practice the realized throughput of a wireless network is less than a radio's theoretical maximum transmission rate.

5. What is the approach used to find link stability in ABR ? [CO3-L1]

A mobile ad hoc network (MANET) is a group of autonomous mobile nodes that wirelessly communicate with each other to form a wireless dynamic topology network. It works without requiring any centralized pre-existing administration units (infrastructure less network). There are many studies that focus on improving source-destination route stability and lifetime by modifying the existing MANET routing protocols. In this paper, a fuzzy-based approach is proposed to enhance the ad hoc on-demand distance vector (AODV) reactive routing protocol's performance by selecting the most trusted nodes to construct the route between the source and destination nodes

.

6. List the major classification of routing protocol for ad hoc wireless network. [CO3-L2]

I. ABR - Associativity-Based Routing[1]
II. Ad hoc On-demand Distance Vector(AODV) (RFC 3561)[2]
III. Dynamic Source Routing (RFC 4728)[3][4]
IV. Flow State in the Dynamic Source Routing[5]
V. Power-Aware DSR-based[6]

7. Based on routing information update mechanism how the routing protocols are classified? [CO3-L2]

Based on the routing information update mechanism: a – Table driven routing protocols: - Periodic exchange of routing information. => "high routing overhead" (-ve) - Each node maintains its own routing table. => "Fast to find routes" (+ve) b – On-demand routing protocols: - No periodic exchange of routing information. => "routing overhead grows according to actual needs" (+ve) - route is found when only required. => "route setup takes more time" (-ve) c – Hybrid: e.g. a (at a defined local zone) + b (among zones)

8. How does energy aware routing work? (May/ June 2012) [CO3-L1]

In some networks, routing is complicated by the fact that no single entity is responsible for selecting paths; instead, multiple entities are involved in selecting paths or even parts of a single path. Complications or inefficiency can result if these entities choose paths to optimize their own objectives, which may conflict with the objectives of other participants.

A classic example involves traffic in a road system, in which each driver picks a path that minimizes their travel time. With such routing, the equilibrium routes can be longer than optimal for all drivers. In particular, Braess' paradox shows that adding a new road can *lengthen* travel times for all drivers.

9. List the classification of routing protocols based on the routing information update mechanism. [CO3-L2]

1. This network is a static densely deployed network. It means a large number of sensor nodes are densely deployed in a two-dimensional geographic space, forming a network and these nodes do not move any more after deployment.
2. There exists only one base station, which is deployed at a fixed place outside A.
3. The energy of sensor nodes cannot be recharged.
4. Sensor nodes are location-unaware, i.e. a sensor node cannot get its location information through other mechanism such as GPS or position algorithms.
5. The radio power can be controlled, i.e., a node can vary its transmission power depending on the distance to the receiver [5]. For instance, Berkeley Motes [12] have in total 100 power levels.

10. List the approaches for power aware routing protocol. [CO3-L2]

- ✓ To maximize network throughput
- ✓ To maximize network lifetime
- ✓ To minimize delay.

11. Based on the use of temporal information for routing, how the routing protocols are classified? [CO3-L2]

1) Minimal Energy Consumption per Packet
2) Maximize Network Connectivity
3) Minimum Variance in Node Power Levels
4) Minimize Maximum Node

12. Based on the routing topology how the routing protocols are classified? [CO3-L1]

- ✓ Routing information update mechanism.
- ✓ Use of temporal information for routing
- ✓ Routing topology
- ✓ Utilization of specific resources.

13. What is the need for power management in ad hoc network? [CO3-L2]

Online adaptation of the keep-alive timers: From the simulation results, fixed keep-alive timers can sometimes be wasteful in the case of on-off traffic sources. We are currently working on the theoretical analysis of the trade-off between energy, latency and throughput. This will provide a basis for adapting the keep-alive timer values to the arrival pattern in the network.

14. List some examples of table driven routing protocols. [CO3-L1]

Table-**driven Routing Protocols**: Keeps **routing tables** at each node, which is updated periodically, and the **routing** procedure is based on this data. One **routing-table driven** method is the next-hop model. In this model, each node keeps the best next-hop node information for all destination nodes in the network.

15. List the advantages and disadvantages of DSDV routing protocols. [CO3-L1]

Advantages

- ■ Simple (almost like Distance Vector)
- ■ Loop free through destination seq. numbers
- ■ No latency caused by route discovery

Disadvantages

- ■ No sleeping nodes
- ■ Overhead: most routing information never used

16. What is hybrid routing protocol? [CO3-L1]

Hybrid Routing Protocol (HRP) is a network **routing protocol** that combines Distance Vector **Routing Protocol** (DVRP) and Link State **Routing Protocol**(LSRP) features. HRP is used to determine optimal network destination routes and report network topology data modifications.

17. Mobility of nodes in a sparsely populated mobile ad ho network is less. What is the choice between proactive routing protocol and reactive routing protocol? [CO3-L1]

Ad-hoc networks and their mobility features. Furthermore, to identify the performance challenges for routing protocols in such networks. Implementing& analyze the existing DSDV and DSR, AODV routing protocols in ns2 .Comparison regarding performance of different routing protocols for the same set of performance metrics in mobile nodes network. For this purpose tabulated results are shown. This comparison helps to see which routing protocol performs best in mobile nodes network. Comparing the performance of three protocols under following metrics

(i) Packets loss (ii) Average Delay (iii) Throughput

18. List the types of on-demand routing protocols. [CO3-L2]

1. Advantage depends on number of other nodes activated.
2. Reaction to traffic demand depends on gradient of traffic volume.ads Examples of hybrid algorithms are:

- ZRP (Zone Routing Protocol)[7] ZRP uses IARP as pro-active and IERP as reactive component.
- ZHLS (Zone-based Hierarchical Link State Routing Protocol)

19. List the types of hybrid routing protocols. [CO3-L1]

- CBRP (Cluster Based Routing Protocol)
- FSR (Fisheye State Routing protocol)
- Order One Network Protocol; Fast logarithm-of-2 maximum times to contact nodes. Supports large groups.
- ZHLS (Zone-based Hierarchical Link State Routing Protocol)

20.How on-demand routing protocols differ from on-demand routing protocols? [CO3-L2]

The characteristic feature of Hierarchical State Routing (HSR) [Iwata99]is multilevel clustering and logical partitioning of mobile nodes. The network is partitioned into clusters and a cluster-head elected as in a cluster-based algorithm. In HSR, the cluster-heads again organize themselves into clusters and so on. The nodes of a physical cluster broadcast their link information to each other. The cluster-head summarizes its cluster's information and sends it to neighboring cluster-heads via gateway

Part B

1. With suitable trace, explain the route establishment in location aided routing. [CO3-H3]

Mobfie ad hw networks consist of wireless mobtie hosts that communicate with each other, in the absence of a fixed ittfrastmctttre.1 Rotttesbetween two hosts in a Mobtie Ad hoc NE~ork ~ANE~ may consist of hops through other hosts in the network [7]. Host mobitity can cause frequent unpredictable topology changes. Therefore, the task of finding and maintaittiig routes in MANET is nontrivial. Many prot~ols have been proposed for mobile ad hm networks, with the god of achieving efficient routing [6,9, 11, 12, 14, 16,17,18,21,23,24, 28]. These dgonthrns differ in the approach used for searehirtg a new route artdor modl~lng a known route, when hosts move. h this paper, we suggest an approach to decreme overhead of route discovery by tstilrtg Iwation information for mobile hosts. Such Ioeation information maybe obtained using the global ~sitioning system (GPS) [10, 22]. We demonstrate how location information may be used by means of two bcation-AZed Routing &AR) protocols for route discovery. The LAR protocols use Iocation information (which maybe out of date, by the time it is used) to reduce the search space for a desired route. Ltiting the search space restdts in fewer route discovery messages

Design of routing protocols is a crucial problem in mobtie ad hoc networks [7, 25], and several routing dgorithnts have been developed (e.g., [6, 9, 11, 12, 14, 16, 17, 18,21,23,24, 28]). One desirable qttrdhative property of a routing protocol is that it should adapt to the tic patterns [8]. Johnsonmtd Mrdtz [15, 16] point out that conventional routing protocols are insufficient for ad hoc networks, since the amount of routing related traffic may waste a large portion of the wireless bandwidth, especially for protocols that use periodic updates of routing tables. They proposed using Dynamic Source Routing @SR), which is based on ondenrand route discovery. A number of protocol optimization are dso proposed to reduce the route discovety overhead. Perkins and Royer [23] present the AODV (Ad hoc On Demand Distance vector routing) protocol that dso uses a demand-driven route establishment procedure. More recent TORA ~empodly-Ordered Routing Algorithm) [21] is designed to minimize reaction to topological changes by locWmg routing-related messages to a small set of nodes near the change. Hass and Pearlmrm [12] attempt to combine proactive and reactive approaches in the Zone Routing Protocol ~RP), by inhiathtg route discove~ phase ondemartd, but timits the scope of the proactive procedure ody to the initiator's Iocrd neighborhood. Also, ZRP Emits @pology update propagation to the neighborhood of the chmge. There is a recent approach for comparative performance evahtation of seved routing protocols proposed in MANET [26]. The existing MANET routing rdgorithms do not take into account the physical location of a destination node.

h this paper, we propose two dgonthtns to reduce route discovery overhead using location information. Similar ideas have been applied to develop selective paging for cellular PCS @ersortd Contnurrtication Service) networks [4]. h selective paging, the system pages a selected subset of cells close to the last reported l~ation of a mobfle hos~ This allows the location tracking cost to be decreased. We propose and evahrate art artdogous approach for routing in MANET. Metricom is a packet radio system using location information for the muting purpose [19]. The Metricom network infrastructure consists of fixed base stations whose precise location is determined using a GPS receiver at the time of instigation. Metricom uses a geographicdy based routing scheme to defiver packets between base stations. Thus, a packet is forwarded one hop closer to its firtrd destination by comparing the location of packet's destination with the location of the node currently holding the packe~ h a survey of potential applications of GPS, Dorntnety and Jairt [10] briefly suggest use of location information in ad hoc networks, though they do not elaborate on how the information maybe used. Other researchers have dso suggested that location information should be used to improve (qttditatively or quantitatively) performance of a mobfle computing system [27, 29]. A routing and addressing method to integrate the concept of physicrd location @eographic coordinates), into the current design of the hteme~ has been investigated in [13, 20]. 3 LocatiomAided Routing(LAR) Protocols 3.1 Route Discovery Using Flooding h this paper, we explore the possibfity of using location information to improve performrmce of routing protocols for ~NET. As Nus@tion, we show how a route discovery protocol based on jooding can be improved. The route discovery dgoriti using flooding is described next (Ms rdgoriti is stiar to Dynamic Source Routing [15, la). men a node S needs to find a route to node D, node S broadcasts a route request message to W its neighbors2 - hereafter, node S wti be referred to as the sender and node D as the destimtion. A node, say X, on receiving a route request message, compares the desired destination with its own identier. E there is a match, it means that the request is for a route to itse~ fi.e., node ~. Otherwise, node X broadcasts tie request to its neighbom -to avoid redundmt ~smissions of route requests, anode X ody broadcasts a pticular route request once (repeated reception of a route request is detected using sequence numbers). Figure 1 ~ustrates this dgoriti. k this figure, node S needs to determine a route to node D. Therefore, node S broadcasts a route request to its neighbors. men nodes B and C receive the route reques~ they forward it to d their neighbors. men node X receives the route request from B, it forwards the request to its neighbom. However, when node X receives the same route request horn C, node X simply discards the route reques~ c + s \ route request A / —o D K F— B E Figure 1: Mustition of flooding As the route request is propagatedto various nodes, the pathfolIowed by the request is included in the route request packet Using the above flooding rdgorithm, provided that the intended destination is reachable from the sender, the destination shodd eventudy receive a route request message. On receiving tie route reques~ the destination responds by sending a route reply message to the sender- the route reply message fo~ows a path that is obtained by reversing the path foflowed by the route request received by D (the route request message includes tie path traversed by the request). It is possible that the destination W not receive a route request message (for instance, when it is unreachable from the sender, or route requests are lost due to transmission errors). h such cases, the sender needs to be able to re-initiate route discovery. Therefore, when a sender initiates route discovery, it sets a timeou~ E during tie timeout interval, a route reply is not received, then a nW route discovery is initiated (the route request messages for this route discovery W use a different sequence number than the previous route discovery - rec~ that sequence numbers are useful to detect multiple receptions of the same route request). Timeout may occur if the destination does not receive a route reques~ or if the route reply message from tie destination is 10SL Route discovery is initiated either when the sender S detects that a previously determined route to node D is broken, or ifS does not know a route to the destination. k our implementation, we assume that node S can know that the route is broken ordy if it attempts to use the route. men node S sends a data packet along a particular route, anode along that path returns a mute error message, if the next hop on the route is broken. men node S receives the route error message, it initiates route discovery for destination D. men using the above dgori~ observe that the route request wodd reach every node that is reachable tim node S @tentiWy, W nodes in the ad hoc network). Using location information, we attempt to reduce the number of nodes to whom route request is propagated. Dynamic source routing @SR) [15, la and adhoc on~emand distance vectorrouting (AOD~ [23] protocols proposedpreviously are both based on vtiations of flooding. DSR and AODV *O use some optimization - seved of these opdrnizations as we~ as other optirnizations suggested in this paper can be used in conjunction with the proposed algorithms. However, for sirnpficity, we Mt our discussion to the basic flooding dgonthm, and location-aided route discove~ based on “tited’ flooding. 3.2 Prefiminarie6 Location Information The proposed approach is termed bcation-Aided Routing @R), as it makes use of location information to reduce routing overhead. hcation information usedin the LAR protocol maybe provided by the Global Positioning System (GPS) [2,3, 10, 22]. Whh the avdabfity of GPS, it is possible for a mobfle host to know its physical locations. k retity, position information provided by GPS includes some amount of error, which is the difference between GPS-crdcdated coordinates and the red coordinates. For instance, NAVSTAR Global Positioning System has positional accuracy of about 50-100 meters and Differential GPS offers accuracies of a few meters [2, 3]. k our initird discussion, we assume that each host knows its current location preckely (i.e., no error). However, the ideas suggested here crm dso be appfied when the location is known ody approximately - the Performance Evaluation section considers this possibtity. k this paper, we assume that the mobfie nodes are moving in a two%ensiond plane. Expected Zone and Requmt Zone Expected Zone: Consider a node S that needs to tid a route to node D. Assume that node S knows that node D was at location L at time to, and that the current time is tl. Then, the "~ected zone" of node D, tim the viewpoint of node S at time ti, is tie region that node S expects to contain node D at time tl.Node S can determine the expected zone based on the knowledge that node D was at location L at time to.For instance, if node S knows that node D travels with average speed v, then S may assume that the expected zone is the circtiar region of radius v(tl – to), centered at location L (see Figure 2(a)). K actual speed happens to be larger than the average, then the desdnation may actudy be outside the expected zone at time tl.Thus, expected zone is otiy an estimate made by node S to determine a region that potenti~y contains D at time tl. E node S does not know a previous location of node D, then node S cannot reasonably determine the expected zone - in this case, the entire region that may potentidy be occupied

hoc network is assumed to be the expected zone. h tis case, our dgonthrn reduces to the b=ic flooding dgonthm. h gened, having more information regarding mobfity of a destination node, can resdt in a stier expected zone. For instance, if S knows that destination D is moving north, then the circtiar expected zone in Figure 2(a) can be reduced to a semicircle, as in Figure 20). L Figure 2 &amples of qectedzone Request Zone: Agti, consider node S that needs to determine a route to node D. The proposed LAR algorithms use flooding with one mtication. Node S defines (imphcifly or expticidy) a request zone for the route reques~ A node forwards a route request only r~it belongs to tie request zone (urdike the flooding algorithm in Section 3.1). To increase the probabtity that the route request W reach node D, the request zone should include the ~ected zone (described above). Additional, the request zone may dso include other regions around the request zone. There are two reasons for m ● When the expected zone does not include host S, a pati tim host S to host D must include hosts ou~ide the expected zone. Therefore, additiond region must be included in the request zone, so that S and D both belong to the request zone (for instance, as shown in Figure 3(a)). ● The request zone in Figure 3(a) includes the expected zone tim Figure 2(a). k this an adequate request zone? h the example in Figure 3@), M paths from S to D include hosts that aze outside the request zone. Thus, there is no gumtee that a path can be found consisting ody of the hosts in a chosen request zone. Therefore, if a route is not discovered within a suitable timeout period, our protocol Wows S to initiate a new route discoveV with an expanded request zone – in our sirmdations, the expanded zone includes the entire network space. hs this even~ however, the latency in determiningg the route to D M be longer (as more than one round of route request propagation WMbe needed). Note that the probabihty of finding a path on tie tit attempt) can be increased by increasing the size of the initial request zone (for instance, see Figure 3(c)). However, route discovery overhead dso increases with the size of the request zone. Thus, there exisk a &de-off between latency of route determination and tie message overhead. 3.3 Determining Membership of Request Zones As noted above, our LAR dgoriti are essentidy identicd to flooding, with the modification that anode that is@ in the request zone does not forward a route request to its neighbom.4 Thus, im- '~~ ti~ h tie floodingtigoriti, a node fomards a route rquest if it hm not rmived the rquest &fore md it is not tie intendeddestination. 68 (.) Figure 3: Request zone An edge bween two nodes means that they are neighbors plementing LAR dgonti requires that a node be able to determine if it is in the request zone for a partictiar route request - the two LAR dgorhhms presented here differ in the manner in which this determination is made. L~ Scheme 1 Gur first scheme uses a request zone that is rectan~azin shape (refer to Figure 4). Assume that node S knows that node D was at location (X~, Y~) at time tO. At dme ti,node S initiates anew route discove~ for destination D. We assume that node S dso bows the average speedu with which D can move. Using this, node S defies the expected zone at time tlto be the circle of radius R=v(tl – to) centered at location (X~, Y~). hs our fit LAR dgonthrn, we define the request zone to be the sdest rectangle that includes current location of S and tie expected zone (the circtiar region defied above), such that the sides of the rectangle are ptiel to the X and Y axes. h Figure 4(a), the request zone is the rectangle whose comers are S, A, B and C, whereas in Figure 4@), the rectangle has comers at points A, B, C and G - note tha~ in this figure, current location of node S is denoted as (X,, Y,). me source node S can thus determine the four comers of the expected zone. S includes their coordinates with the route request message transmitted when inhiating route discovery. When a node receives a route reques~ it discards the request if the node is not within the rectangle specfied by the four comers included in the route reques~ For instance, in Figure 4(a), if node I receives the mute request from anotier node, node I forwards the request to its neighbors, because I determines that it is within the rectan@ar request zone. However, when node J receives the route reques~ node J discards the reques~ as node J is not within the request zone (see Figure 4(a)). When node D receives the route request message, it repfies by sending a route reply message (as in tie flooding rdgorithm). However, in case of LAR, node D includes its current location and current time in the route reply message. When node S receives this route reply message (ending its route discove~), it records the location of node D. Node S can use this information to determine the request zone for a fiture route discovery. ~t is dso possible for D to include its current speed, or average speed over a recent time interval, with the route reply message. This information codd be used in a future route discovery. hour simtiations, we assume that d nodes know each other's average speed.) Size of the request zone: Note that the size of the rectan~ar request zone above is proportional to (i) average speed of movement u, and (ii) time elapsed since the last known location of the destination was recorded. hs our implementation, the sender comes

● me coordinates (Xd, Yd) are rdso included with the route request men anode I receives the route request horn sender node S, node I crdctiates its distance horn location (X~, Y~),denoted as DISTi, and ● ● For some parameter 6, if DIST, + 6 ~ DIST~, tien node I forwards the request to its neighbors. men node I forwards the route reques~ it now includes DIST; md (X~, Yd) in the mute request @e., it replaces the DIST, value received in tie route request by DIST~, before forwarding the route request). Else DIST, + 8< DISTi. h this case, node I discardsthe route request men some node J receives the mute request (originated by node S) from node I, it appfies a criteria stiar to abov~ K node J has received this request previously, it discards the request Otierwise, node J cdcdates its distance from (X~, Yd), denoted as DISTj. NOW, ● me mute request received from I includes DISTi. KDISTi +6 ~ DISTj, then node J forwards the request to its neighbors (urdess node J is the destination for the route request). Before forwarding tie reques~ J replaces the DISTi value in the route request by DISTj. ● Else DISTi + J < DISTj. h thiscase,node J diSCNdS he requesL ~us, a node J forwards a route request forwarded by I (originated by node S), if J is “at most d farthef’ horn (X~, Yd) than node I. For the purpose of petiorrnance evaluation, we use 6 = O in the next section. Non-zero J maybe used to trade~ff the probabfity of finding a route on the tit attempt with the cost of tiding the route. Non-zero 6 may dso be appropriate when location error is non-zero, or when the hosts am Wely to move si@cant distances during the time required to petionn route discove~. Figure 5 Nustrates the difference between the two L~ schemes. Consider Figure 5(a) for L~ scheme 1: men nodes I and K receive tie route request for node D (originated by node S), they forward the route reques~ as both I and K are within the rectan@ar request zone. On the otier hand, when node N receives tie route requesg it discards the reques~ as N is outside the rectan~ar request zone. Now consider Figure So) for L~ scheme 2 (assume 6 = O): Men nodes N and I receive the route request from node S, both forward the route request to heir neighbors, because N and I are both closer to (Xd, Y~) than node S. men node K receives tie route request from node I, node K discards the route reques~ as K is farther tim (Xd, Yd) than node I. Observe that nodes N and K take different actions when using the two L~ schemes. Error in Location Estimate k the above, we assume that each node knows its own location accurately. However, in reti~ Weremaybe some error in the estimated location. Let e denote the maximum error in the coordinates estimated by a node. ~us, if a node N betieves that it is at location (X., Y=), then the actual location of node N may be anywhere in the circle of radius e centered at (Xn, Yn). h the next section, we W refer to e as location error. h tie above L~ schemes, we assume that node S obtied the location (Xd, Y~) of node D at time to, tim node D @rhaps in the route reply message during the previous route discove~). ~us, node S does not know the acturd location of node D at time to - the actual location is somewherein the circle of radius e centered at (Xd, Y~)

2. Device a pseudo code that present various steps involved in neighbour Degree- Based preferred link algorithm. [CO3-H1]

A mobile ad hoc network is a network wherein a pair of nodes communicates by sending messages either over a direct wireless link, or over a sequence of wireless links including one or more intermediate nodes. Direct communication is possible only between pairs of nodes that lie within one another's transmission radius. Wireless link \failures" occur when previously communicating nodes move such that they are no longer within transmission range of each other. Likewise, wireless link \formation" occurs when nodes that were too far separated to communicate move such that they are within transmission range of each other. Characteristics that distinguish ad hoc networks from existing distributed networks include frequent and unpredictable topology changes and highly variable message delays. These characteristics make ad hoc networks challenging environments in which to implement distributed algorithms. Past work on modifying existing distributed algorithms for ad hoc networks includes numerous routing protocols (e.g., [8,9,11,13,16,18,19,22{24]), wireless channel al location algorithms (e.g., [14]), and protocols for broadcasting and multicasting (e.g., [8,12,21,26]). xed wired networks that share some characteristics of ad hoc networks, since failure and repair of nodes and links is unpredictable in both cases. Research on dynamic networks has focused on total ordering [17], end-to-end communication, and routing (e.g., [1,2]). Existing distributed algorithms will run correctly on top of ad hoc routing protocols, since these protocols are designed to hide the dynamic nature of the net- work topology from higher layers in the protocol stack (see Figure 1(a)). Routing algorithms on ad hoc net- works provide the ability to send messages from any node to any other node. However, our contention is that eÆciency can be gained by developing a core set of distributed algorithms, or primitives, that are aware of the underlying mobility in the network, as shown in Figure 1(b). In this paper, we present a mobility aware distributed mutual exclusion algorithm to illustrate the layering approach in Figure 1(b). The mutual exclusion problem involves a group of processes, each of which intermittently requires access to a resource or a piece of code called the critical section (CS). At most one process may be in the CS at any given time. Providing shared access to resources through mutual exclusion is a fundamental problem in computer science, and is worth considering for the ad hoc environment, where stripped-down mobile nodes may need to share resources. Distributed mutual exclusion algorithms that rely on the maintenance of a logical structure to provide order and eÆciency (e.g., [20,25]) may be ineÆcient when run 2 User Applications Distributed Primitives Routing Protocol Ad Hoc Network (b) Distributed Primitives Routing Protocol Ad Hoc Network User

Applications (a) Figure 1. Two possible approaches for implementing distributed primitives. in a mobile environment, where the topology can potentially change with every node movement. Badrinath et al.[3] solve this problem on cellular mobile networks, where the bulk of the computation can be run on wired portions of the network. We presentamutual exclusion algorithm that induces a logical directed acyclic graph (DAG) on the network, dynamically modifying the logical structure to adapt to the changing physical topology in the ad hoc environment. We then present simulation results comparing the performance of this algorithm to a static distributed mutual exclusion algorithm running on top of an ad hoc routing protocol. Simulation results indicate that our algorithm has better average waiting time per CS entry and message complexity per CS entry no greater than the cost incurred by a static mutual exclusion algorithm running on top of an ad hoc routing algorithm when nodes are mobile. The next section discusses related work. In Section 3, we describe our system assumptions and define the problem in more detail. Section 4 presents our mutual exclusion algorithm. We present a proof of correctness and discuss the simulation results in Sections 5 and 6, respectively. Section 7 presents our conclusions. 2. Related Work Token based mutual exclusion algorithms provide access to the CS through the maintenance of a single token that cannot simultaneously be present at more than one node in the system. Requests for CS entry are typically directed to whichever node is the current token holder. Raymond [25] introduced a token based mutual exclusion algorithm in which requests are sent, over a static spanning tree of the network, toward the token holder; this algorithm is resilient to non-adjacent node crashes and recoveries, but is not resilient to link failures. Chang et al.[7] extend Raymond's algorithm by imposing a logical direction on a sufficient number of links to induce a token oriented DAG in which, for every node i, there exists a directed path originating at i and terminating at the token holder. Allowing request messages to be sent over all links of the DAG provides resilience to link and site failures. However, this algorithm does not consider link recovery, an essential feature in a system of mobile nodes. Dhamdhere and Kulkarni [10] show that the algorithm of [7] can suffer from deadlock and solve this problem by assigning a dynamically changing sequence number to each node, forming a total ordering of nodes in the system. The token holder always has the highest sequence number, and, by defining links to point from a node with lower to higher sequence number, a token oriented DAG is maintained. Due to link failures, a node i that wants to send a request for the token may find itself with no outgoing links to the token holder. In this situation, i floods the network with messages to build a temporary spanning tree. Once the token holder becomes part of such a spanning tree, the token is passed directly to node i along the tree, bypassing other requests. Since priority is given to nodes that lose a path to the token holder, it seems likely that other requesting nodes could be starved as long as link failures continue. Also, flooding in response to link failures and storing messages for delivery after link recovery make this algorithm ill-suited to the highly dynamic ad hoc environment. Our token based algorithm combines ideas from several papers. The partial reversal technique from [13], used to maintain a destination oriented DAG in a packet radio network when the destination is static, is used in our algorithm to maintain a token oriented DAG with a dynamic destination. Like the algorithms of [25], [7], and [10], each node in our algorithm maintains a request queue containing the identifiers of neighboring nodes from which it has received requests for the token. Like [10], our algorithm totally orders nodes. The lowest node is always the current token holder, making it a \sink" toward which all requests are sent. Our algorithm also includes some new features. Each node dynamically chooses its lowest neighbor as its preferred link to the token holder. Nodes sense link changes to immediate neighbors and reroute requests based on the 3 status of the previous preferred link to the token holder and the current contents of the local request queue. All requests reaching the token holder are treated symmetrically, so that requests are continually serviced while the DAG is being re-oriented and blocked requests are being rerouted. 3. Definitions The system contains a set of n independent mobile nodes, communicating by message passing over a wireless network. Each mobile node runs an application process and a mutual exclusion process that communicate with each other to ensure that the node cycles between its REMAINDER section (not interested in the CS), its WAITING section (waiting for access to the CS), and its CRITICAL section. Assumptions1 on the mobile nodes and network are: 1. the nodes have unique node identifiers, 2. node failures do not occur, 3. communication links are bidirectional and FIFO, 4. a link-level protocol ensures that each node is aware of the set of nodes with which it can currently directly communicate by providing indications of link formations and failures, 5. incipient link failures are detectable, providing reliable communication on a per-hop basis, and 6. partitions of the network do not occur. The rest of this section contains our formal definitions. We explicitly model only the mutual exclusion process at each node. Constraints on the behavior of the application processes and the network appear as conditions on executions. The system architecture is shown in Figure 2. We assume the node identifiers are 0; 1;:::;n 1. Each node has a (mutual exclusion) process, modeled as a state machine, with the usual set of states, some of which are initial states, and a transition function. Each state contains a local variable that holds the node identi er and a local variable that holds the current neighbors of the node. The transition function is described in more detail shortly. 1 See Section 7 for a discussion of relaxing assumption 6. Application Process Mutual Exclusion Process Network node i RequestCS ReleaseCS LinkUp Send(m) LinkDown Recv(m) EnterCS Figure 2. System architecture. A configuration describes the instantaneous state of the whole system; formally, it is a set of n states, one for each process. In an initial configuration, each state is an initial state and the neighbor variables describe a connected undirected graph. A step of the process at node i is triggered by the occurrence of an input event. Input events are:

RequestCSi : the application process on node i requests access to the CS, entering its WAITING section. ReleaseCSi: the application process on node i releases its access to the CS, entering its REMAINDER section. Recvi(j; m): node i receives message m from node j. LinkUpi(l): node i receives notication that the link l incident on i is now up. LinkDowni(l): node i receives notication that the link l incident on i is now down. The e ect of a step is to apply the process' transition function, taking as input the current state of the process and the input event, and producing as output a (possibly empty) set of output events and a new state for the process. Output events are: EnterCSi : the mutual exclusion process on node i informs its application process that it can enter the CRITICAL section. Sendi(j; m): node i sends message m to node j. The only constraint on the state produced by the transition function is that the neighbor set variable of i must be properly updated in response to a LinkUp or LinkDown event. 4 RequestCSi , EnterCSi, and ReleaseCSi are called application events, while Sendi , Recvi , LinkUpi , and LinkDowni are called network events. An execution is a sequence of the form C0, in1, out1, C1, in2, out2, C2;:::, where the Ck 's are congurations, the ink's are input events, and the outk's are sets of output events. An execution must end in a conguration if it is nite. A positive real number is associated with each ini , representing the time at which that event occurs. An execution must satisfy a number of additional conditions, which we now list. The rst set of conditions are basic \syntactic" ones. C0 is an initial conguration. If ink occurs at node i, then outk and i's state in Ck are correct according to i's transition function operating on ink and i's state in Ck1. The times assigned to the steps must be nondecreasing. If the execution is innite, then the times must increase without bound. At most one step by each process can occur at a given time. The next set of conditions require the application process to interact properly with the mutual exclusion process and to give up the CS in nite time. If ink is RequestCSi , then the previous application event at node i (if any) is ReleaseCSi. If ink is ReleaseCSi, then the previous application event at node i must be EnterCSi. If outk includes EnterCSi , then there is a following ReleaseCSi. The remaining conditions constrain the behavior of the network to match the informal description given above. First, we consider the mobility notication. LinkUpi(l) occurs at time t if and only if LinkUpj (l) occurs at time t, where l joins i and j. Furthermore, LinkUpi(l) only occurs if j is currently not a neighbor of i (according to i's neighbor variable). The analogous condition holds for LinkDown. A LinkDown never disconnects the graph. Finally, we consider message delivery. There must exist a one-to-one and onto correspondence between the occurrences of Sendj (i; m) and Recvi(j; m), for all i, j and m. This requirement implies that every message sent is received and the network does not duplicate or corrupt messages nor deliver spurious messages. Furthermore, the correspondence must satisfy the following: If Sendi(j; m) occurs at some time t, then the corresponding Recvj (i; m) occurs at some later time t 0 , and the link connecting i and j is continuously up between t and t 0 . This implies that a LinkDown event for link l cannot occur if any messages are in transit on l. Now we can state the problem formally.In every execution, the following must hold: If outk includes EnterCSi , then the previous application event at node i must be RequestCSi . I.e., CS access is only given to requesting nodes. Mutual Exclusion: If outk includes EnterCSi , then any previous EnterCSj event must be followed by a ReleaseCSj prior to outk . No Starvation: If there are only a nite number of LinkUpi and LinkDowni events, then if ink is RequestCSi , then there is a following EnterCSi. For the last condition, the hypothesis that link changes cease is needed because an adversarial pattern of link changes can cause starvation.

3. How is routing table constructed in fisheye state routing protocol? Explain in detail. [CO3-H2]

As the wireless and embedded computing technologies continue to advance, increasing numbers of small size and high performance computing and communication devices will be capable of tetherless communications and ad hoc wireless networking. An ad hoc wireless network is a selforganizing and self-configuring network with the capability of rapid deployment in response to application needs. An important characteristic which sets ad hoc networks apart from cellular networks is the fact that they do not rely on a fixed infrastructure. Ad hoc networks are very attractive for tactical communication in military and law enforcement. They are also expected to play an important role in civilian forums such as convention centers, conferences, and electronic classrooms. Mobility, potentially very large number of mobile nodes, and limited resources (e.g., bandwidth and power) make routing in ad hoc networks extremely challenging. The routing protocols for ad hoc wireless networks have to adapt quickly to the frequent and unpredictable changes of topology and must be parsimonious of communications and processing resources. ✁ This work was supported in part by NSF under contract ANI-9814675, in part by DARPA under contract DAAB07-97-C-D321 and in part by Intel. In this paper, we introduce a new routing scheme for ad hoc wireless networks. It is a link state based routing protocol which is adapted to the wireless ad hoc environment. The rest of the paper is organized as follows. In section 2, we survey the existing wireless routing schemes. We describe the Fisheye State Routing (FSR) in section 3. Section 4 presents the performance results and we conclude our paper in section 5. 2 Brief Review of Routing Protocols Existing wireless routing schemes can be classified into three categories according to their design philosophy: (a) distance vector based; (b) link state based; (c) on demand. Historically, the first type of routing scheme used in early packet networks such as the ARPANET was the distance vector type. The main advantages of the distance vector approach are simplicity and computation efficiency. However, this approach suffers from slow convergence and tendency of creating routing loops. While several approaches were proposed which solve the looping problem [17, 15], none of them overcome the problem of slow convergence. The solutions to both convergence and looping come in the form of the Link State (LS) approach. LS is the preferred scheme for wired nets (e.g., Internet [14] or ATM [1]). In Link State, global network topology information is maintained in all routers by the periodic flooding of link state updates by each node. Any link change triggers an immediate update. As a result, the time required for a router to converge to the new topology is much less than in the distance vector approach. Due to global topology knowledge, preventing routing loop is also easier. Unfortunately, as Link State relies on flooding to disseminate the update information, excessive control overhead may be generated, especially when mobility is high and frequent updates are triggered. In addition, the small update packets make for inefficient use of the wireless MAC layer. When wireless, ad hoc network size and mobility increase (beyond certain thresholds), current proactive routing schemes (i.e., the dis- tance vector and link state) become infeasible since they will consume a large part of network capacity and node processing power to transmit update control messages just to keep up with the topology changes. The most recent addition to the family are the on demand routing schemes. These have been specifically introduced in order to overcome some limitations of the proactive protocols in mobile environments. Examples include AODV [18], TORA [16], DSR [6], ABR [20]. The basic idea behind these reactive protocols is that a node discovers a route in an “on demand” fashion, namely, it computes a route only when needed. In on demand schemes, query/response packets are used to discover (possible more than) one route to a given destination. These control packets are usually smaller than the control packets used for routing table updates in proactive schemes, thus causing less overhead. However, since a route has to be entirely discovered prior to the actual data packet transmission, the initial search latency may degrade the performance of interactive applications (e.g., distributed database queries). Moreover, it is impossible to know in advance the quality of the path (e.g., bandwidth, delay etc) prior to call setup. Such a priori knowledge is very desirable in multimedia applications, since it enables more effective call acceptance control. If the route breaks down because of mobility, a packet may need multiple route discoveries on the way to destination. Route caching becomes ineffective in high mobility. Since flooding is used for query dissemination and route maintenance, on demand routing tends to become inefficient when traffic load and mobility are high and network size grows large [10]. A recent proposal which combines on demand routing and conventional routing is Zone Routing Protocol (ZRP) [7, 8]. For routing operation inside a local zone, an arbitrary proactive routing scheme (e.g., distance vector) can be applied. For interzone routing, on demand routing is used. The advantage of zone routing is its scalability, as “global” routing table overhead is limited by zone size. Yet, the benefits of global routing are preserved within each zone. The performance of ZRP is dependent on a key parameter: the zone radius. The choice of radius is determined by network characteristics (e.g, node density, relative node velocity etc.) [8], which dynamically change in ad hoc networks. Moreover, the interzone route discovery packets may loop back into zones already queried. This must be avoided to prevent overhead which can be potentially worse than for flooding based queries [8]. With the availability of GPS [11] technology, any of the previous routing protocols can be assisted by GPS location information. For example, LAR [13] is an on demand protocol similar to DSR but it restricts control packet flooding by using location information. DREAM [3] is a location based proactive scheme. Each node in the network periodically exchanges control packets to inform all the other nodes of its location. Each control packet is assigned a life time based on the geographical distance from the sender. DREAM sends short lived packet more frequently than long lived packets due to the so called distance effect, i.e., the farther two nodes separate, the slower they seem to be moving with respect to each other. The data packet is broadcast to the nodes in the direction of the destination using only location information stored at the sender. 3 Fisheye State Routing (FSR) 3.1 Topology Representation in FSR The ad hoc wireless network is modeled as an undirected graph where ✆ is a set of ✌ ✆ ✌ nodes and ✠ is a set of ✌ ✠ ✌ undirected links connecting nodes in ✆ . Each node has a unique identifier and represents a mobile host with a wireless communication device with transmission range ✍, and an infinity storage space. Nodes may move around and change their speed and direction independently. An undirected link connecting two nodes ✎ and ➼ is formed when the distance between ✎ and ➼ become less than or equal to ✍. Link is removed from ✠ when node ✎ and ➼ move apart, and out of their transmission ranges. In the FSR routing implementation, for each node ✎ , one list and three tables are maintained. They are: a neighbor list, a topology, a next hop table and a distance table is defined as a set of nodes that are adjacent to node ✎ . Each destination ➼ has an entry in table which contains two parts:. denotes the time stamp indicating the time node ➼ has generated this link state information. Similar, for every destination denotes the next hop to forward packets destined to ➼ on the shortest path, while denotes the distance of the shortest path from ✎ to ➼ . Additionally, a weight function, weight, is used to compute the distance of a link. Since min-hop shortest path is the only objective in this paper, this weight function simply returns 1 if two nodes have direct connection, otherwise, it returns ✻. This weight function may also be replaced with other functionsfor routing with different metrics. For instance, a bandwidth function can be used to realize a QoS routing. 3.2 Description of FSR protocol FSR is an implicit hierarchical routing protocol. It uses the “fisheye” technique proposed by Kleinrock and Stevens [12], where the technique was used to reduce the size of information required to represent graphical data. The eye of a fish captures with high detail the pixels near the focal point. The detail decreases as the distance from the focal point increases. In routing, the fisheye approach translates to maintaining accurate distance and path quality information about the immediate neighborhood of a node, with progressively less detail as the distance increases. FSR is functionally similar to LS Routing in that it maintains a topology map at each node. The key difference is the way in which routing information is disseminated. In LS, link state packets are generated and flooded into the network whenever a node detects a topology change. In FSR, link state packets are not flooded. Instead, nodes maintain a link state table based on the up-to-date information received from neighboring nodes, and periodically exchange it with their local neighbors only (no flooding). Through this exchange process, the table entries with largersequence numbers replace the ones with smaller sequence numbers. The FSR periodic table exchange resembles the vector exchange in Distributed Bellman-Ford (DBF) (or more precisely, DSDV [17]) where the distances are updated according to the time stamp or sequence number assigned by the node originating the update. However, in FSR link states rather than distance vectors are propagated. Moreover, like in LS, a full topology map is kept at each node and shortest paths are computed using this map. In a wireless environment, a radio link between mobile nodes may experience frequent disconnects and reconnects. The LS protocol releases a link state update for each such change, which floodsthe network and causes excessive overhead. FSR avoids this problem by using periodic, instead of event driven, exchange of the topology map, greatly reducing the control message overhead. When network size grows large, the update message could consume considerable amount of bandwidth, which depends on the update period. In order to reduce the size of update messages without seriously affecting routing accuracy, FSR uses the Fisheye technique. Fig. 1 illustrates the application of fisheye in a mobile, wireless network. The circles with different shades of grey define the fisheye scopes with respect to the center node (node 11). The scope is defined as the set of nodes that can be reached within a given number of hops. In our case, three scopes are shown for 1, 2 and 2 hops respectively. Nodes are color coded as black, grey and white accordingly. The number of levels and the radius of each scope will depend on the size of the network. The reduction of routing update overhead is obtained by using different exchange periods for different entries in routing table. More precisely, entries corresponding to nodes within the smaller scope are propagated to the neighbors with the highest frequency. Referring to Fig. 2, entries in bold are exchanged most frequently. The rest of the entries are sent out at a lower frequency. As a result, 1 2 3 4 5 6 7 8 9 10 11 13 12 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 34 35 36 Hop=1 Hop=2 Hop>2 9 Figure 1. Scope of fisheye a considerable fraction of link state entries are suppressed in a typical update, thus reducing the message size. This strategy produces timely updates from near stations, but creates large latencies from stations afar. However the imprecise knowledge of the best path to a distant destination is compensated by the fact that the route becomes progressively more accurate as the packet gets closer to destination. As the network size grows large, a “graded” frequency update plan must be used across multiple scopes to keep the overhead low. 0 5 1 2 4 3 0:{1} 1:{0,2,3} 2:{5,1,4} 3:{1,4} 4:{5,2,3} 5:{2,4} 1 0 1 1 2 2 TT HOP 0:{1} 1:{0,2,3} 2:{5,1,4} 3:{1,4} 4:{5,2,3} 5:{2,4} 2 1 2 0 1 2 TT HOP 0:{1} 1:{0,2,3} 2:{5,1,4} 3:{1,4}

4:{5,2,3} 5:{2,4} 2 2 1 1 0 1 TT HOP Figure 2. Message reduction using fisheye The FSR concept originates from Global State Routing (GSR) [5]. GSR can be viewed as a special case of FSR, in which there is only one fisheye scope level and the radius is ✻. As a result, the entire topology table is exchanged among neighbors. Clearly, this consumes a considerable amount of bandwidth when network size becomes large. Through updating link state information with different frequencies depending on the scope distance, FSR scales well

4. Discuss table driven protocols with examples. [CO3-H3]

Reverse Link (RL) Mutual Exclusion Algorithm In this section we rst present the data structures maintained at each node in the system, followed by an overview of the algorithm, the algorithm pseudocode, and examples of algorithm operation. Throughout this section, data structures are described for node i, 0 i n 1. Subscripts on data structures to indicate the node are only included when needed. 4.1. Data Structures status: Indicates whether node is in the WAITING, CRITICAL, or REMAINDER section. Initially, status = REMAINDER. N: The set of all nodes in direct wireless contact with node i. Initially, N contains all of node i's neighbors. myHeight: A three-tuple (h1,h2,i) representing the height of node i. Links are considered to be directed from nodes with higher height toward nodes with lower height, based on lexicographic ordering. E.g., 5 if myHeight1 = (2, 3, 1) and myHeight2 = (2, 2, 2), then myHeight1 > myHeight2 and the link between these nodes would be directed from node 1 to node 2. Initially at node 0, myHeight0 = (0, 0, 0) and, for all i 6= 0, myHeighti is initialized so that the directed links form a DAG in which every node has a directed path to node 0. height[j]: An array of tuples representing node i's view of myHeightj for all j 2 Ni . Initially, height[j] = myHeightj , for all j 2 Ni . In node i's viewpoint, if j 2 N, then the link between i and j is incoming to node i if height[j] > myHeight, and outgoing from node i if height[j] < myHeight. tokenHolder: Flag set to true if node holds token and set to false otherwise. Initially, tokenHolder = true if i = 0, and tokenHolder = false otherwise. next: When node i holds the token, next = i, otherwise next is the node on an outgoing link. Initially, next = 0 if i = 0, and next is an outgoing neighbor otherwise. Q: Queue containing identiers of requesting neighbors. Operations on Q include Enqueue(), which enqueues an item only if it is not already on Q, Dequeue() with the usual FIFO semantics, and Delete(), which removes a specied item from Q, regardless of its location. Initially, Q = ;. receivedLI[j]: Boolean array indicating whether LinkInfo message has been received from node j, to which a Token message was recently sent. Any height information received at node i from a node j for which receivedLI[j] is false will not be recorded in height[j]. Initially, receivedLIi [j] = true for all j 2 Ni . forming[j]: Boolean array set to true when link to node j has been detected as forming and reset to false when rst LinkInfo message arrives from node j. Initially, formingi [j] = false for all j 2 Ni . formHeight[j]: An array of tuples storing value of myHeight when new link to j rst detected. Initially, formHeighti [j] = myHeighti for all j 2 Ni . 4.2. Overview of the RL Algorithm The mutual exclusion algorithm is event-driven. An event at a node i consistsof receiving a message from another node j 6= i, or an indication of link failure or formation from the link layer, or an input from the application on node i to request or release the CS. Each message sent includes the current value of myHeight at the sender. Modules are assumed to be executed atomically. The pseudocode triggered by input events from the application process is shown in Figure 3. When node i requests access to the CS: 1. status := WAITING 2. Enqueue(Q; i) 3. if (not tokenHolder) 4. if (jQj = 1) ForwardRequest() 5. else GiveTokenToNext() When node i releases the CS: 1. if (jQj > 0) GiveTokenToNext() 2. status := REMAINDER Figure 3. Pseudocode triggered by input events from application process. Requesting and releasing the CS: When node i requests access to the CS, it enqueues its own identier on Q and sets status to WAITING. If node i does not currently hold the token and i has a single element on its queue, it calls ForwardRequest() to send a Request message. If node i does hold the token, i can set status to CRITICAL and enter the CS, since it will be at the head of Q. When node i releases the CS, it calls GiveTokenToNext() to send a Token message if Q is non-empty, and sets status to REMAINDER. The pseudocode triggered by network input events is shown in Figures 4 and 5. Request messages: When a Request message sent by a neighboring node j is received at node i, i ignores the Request if receivedLI[j] is false. Otherwise, i changes height[j], and enqueues j on Q if the link between i and j is incoming at

i. If Q is non-empty, and status = REMAINDER,

i calls GiveTokenToNext(), provided i holds the token.

Non-token holding node i calls RaiseHeight()

if the link to j is now incoming and i has no outgoing links or i calls ForwardRequest()

if Q = [j] or if Q is non-empty and the link to next has reversed.

Token messages: When node i receives a Token message from some neighbor j, i sets tokenHolder = true.

Then i lowers its height to be lower than that of the last token holder, node j, informs all its outgoing neighbors of its new height by sending LinkInfo messages, and calls GiveTokenToNext().

Node i also informs j of its new height so that j will know that i received the token. LinkInfo messages: If receivedLI[j] is true

when a LinkInfo message is received at node i from node j,

6 When Request(h) received at node i from node j: // h denotes

j's height when message was sent

1. if (receivedLI[j])

2. height[j] := h // set i's view of j's height

3. if (myHeight < height[j]) Enqueue(Q; j)

```
 4. if (tokenHolder)
5. if ((jQj > 0) and (status = REMAINDER))
6. GiveTokenToNext()
7. else // not tokenHolder
 8. if (myHeight < height[k], 8 k 2 N)
9. RaiseHeight()
10. else if ((Q = [j]) or ((jQj > 0) and (myHeight < height[next])))
11. ForwardRequest() // reroute request When Token(h) received at node i from node j: // h denotes j's height when message was sent
1. tokenHolder := true 2. height[j] := h 3. Send LinkInfo(h.h1, h.h2 1;
i) to all outgoing k 2 N and to j 4. myHeight.h1 := h.h1 5. myHeight.h2 := h.h2 - 1 // lower my height 6. if (jQj > 0) GiveTokenToNext()
7. else next := i When LinkInfo(h) received at node i from node j: // h denotes j's height when message was sent
1. N := N [ fjg
2. if ((forming[j]) and (myHeight 6= formHeight[j]))
3. Send LinkInfo(myHeight) to j
4. forming[j] := false
5. if (receivedLI[j]) height[j] := h
 6. else if (height[j] = h) receivedLI[j]true
 7. if (myHeight > height[j]) Delete(Q; j)
 8. if ((myHeight < height[k], 8k 2 N) and (not tokenHolder))
9. RaiseHeight()
10. else if ((jQj > 0) and (myHeight < height[next]))
11. ForwardRequest() .
```

5.Explain multicast routing algorithms in detail. [CO3-H3]

Multicasling is the ability of a communication network to acccpt a single message from an application and to dclivcr copies of the mcssagc to multiple recipients at different locations. One of the cliallcngcs is to minimize tho amount of network resources employcd by multicasting. To illustrate this point, kt us assume that a vidco server wants to transmit a movie to 1000 recipients (Fig. la). If the server wcre to cmploy 1000 scparate point-to-point connections (e.g., TCP connections), 1000 copies of the movie may have ti) be sent over a singlc link, thus making poor usc of the availablc bandwidth. An efficient implementation of multicasting permits much bettcr nsc of the availeblc bandwidth by transmitting at most onc copy of the movic on cach link in the nctwork, as shown in Fig. lb. Rcccntly, there has bccn a lot of research in the area of multicast communication. Although many excellent surveys and books cxist which cxamine varions aspccts of multicasting [I-61, in thc course of our studies wc have found a need for a tutorial-cum-survcy of the various multicast routing algorithms and their relationship with mnlticast routing protocols. In this work we present a tutorial-cum-survcy of the following two important topics in multicasting: - Multicast routing algorithms * Multicast routing protocols Communication networks can be classificd into two categories: local area networks (LANs) and wide area nctworks (WANs). A LAN spans a small geographical area, typically a single building or a cluster of buildings, while a WAN spans a large geographical area (e.g., a nation). Often, nodes connccted to a LAN communicatc over a broadcast network, while nodes connected to a WAN communicate via a switched network. In a broadcast LAN, a transmission from any one node is received by all the nodes on the network; thus, multicasting is easily implementThis work has heor supporied inpar(/>y ike Nalional Suie,rce Foundution (NSF) under G,owts Nos. NCR 9508238 nnd ANI-9XO52U5. ._ I u&"g a movie" 1000 di&ent users; b) multicasting the movie. (R= slandard router, MR= rnullicast router.) 90 ox~o-xn4~~nn1~io.oo o 2uuo IEEE IEEE Network * JanuaryiFebrualy 2000 6.1, UT, MI, and NY; bj a directedgraph that modelithe WANshown in a). ed on a broadcast LAN. On the other hand, implementing mnlticasting on a switched network is quite challenging; hence, throughout this work, we will focus on the multicasting problem in a WAN which is hascd on a switched network. Today, many multicast applications exist, such as news feeds, filc distribution, interactive games, and videoconferencing, but the implementation of these applications is not necessarily efficient because today's WANs were designed to mainly support point-to-point (unicast) communication, In the future, as multicast applications become more popular and handwidth-intensive, there will emerge a pressing need to provide efficient multicasting sup- . .. . -. . . . . WA I NI CA1 - .. . .- CA2 Tx .. .. ~~ F'aure 3. hi ~~nimi~l.~ uIti .5rwwr iw. MuIiir.o.%r vruui) = I(Al. 'I'X. II. part O~WANS. A WAN consists of nodes (i.e., switches or routers) intcrconnected by communication links. A transmission from a source to a destination is routed through these interconnected nodcs. Fieure 2a shows an examvle of a NY). Cost ofall lih 1. cost of Steiner tree =>. metric and asymmetric. Symmetric links have the same weight in hoth directions. while asvmmetric links have different route of a transmission from ;source to a destination on a WAN.' A WAN can be modeled by a directed graph. Figurc 2h shows a dirccted graph that models the communication network shown in Fig. 2a. A directed graph consists of a set of nodes V and a set of links E. A link connecting node u to nodc U is represented by an ordered tuple (U, U). Nodes in the directed graph reprcsent nodes in the WAN, while links in the directed graph reprcsent communication links in the WAN. (Note that the graph in Fig. 2b is a special one in the sense that if there is a link (U, v), there also cxists a link (U, U); this characteristic, however, is not a necessity in a gencral directed graph.) Communication links in a network may have different properties. For example, a fiber optic communication link may have very large bandwidth compared to a copper wirc communication link. A property of a communication link is represented by a weight of the corresponding link in a graph. For examplc, if the propagation delay of the communication link (CA2,TX) is 1 ms, this information can be represented by assigning a weight equal to 1 to the link (CA2,TX) in Fig. 2h, with the weights of the other links being their corresponding propagation delays in milliseconds. The communication links in Fig. 2 can hc of two typcs: symweights depending'on the diiection. Thus, in Fig. 2b, which shows weights on only four links, the link between nodes CA2 and TX is symmetric, while the link between nodes TX and MD is asymmetric. If all the links in a WAN are symmetric, we can model the WAN by an undirectedgraph, as shown in Fig. 3. In an Undirected graph, the direction of a link is unimportant; hence, a link between node u and node U can he represented by an unordercd tuple (U, v). Traditionally, communication networks have been modcled by undirected graphs. Henceforth in this work, unless othcnvise stated, the term graph will refer to an undirected graph. In unicast (point-to-point) Communication, routing is often treated as the shortest-path problem in graphs. When two nodcs wish to Communicate, a minimum-weight path (shortestpath) connecting the corresponding pair of nodes is selected. In multicasting, a group of more than two nodes (also called the multicast group) wish to communicate with one another. Now, instead of the shortest path, we are interested in the minimumweight tree which spans all the nodcs in the multicast group. In gcneral, differcnt multicast applications havc different requirements. For cxample, a reliable data transfer multicast application, such as software distribution, has very different requirements from a real-time multimedia multicast application, such as nationwide videoconferencing. Thus, it is helpful to classifv multicast communication into two tvnes: IEEE Nehvork IanualyiFebruary 2000 91 ticast group as well as reccivc data from other nodes in the multicast group. The next section discusses multicast routing algorithms. We then study the implementation of multicast routing protocols on the Internet. Note that the current Internet uses IPv4, while the next-generation Internet (NGI) will employ lPv6. Since some topics discussed are specific to IPv4, they are not applicable to the NGI, although the general principles discussed will still be applicable. On the other hand, the subscctions on multicast routing algorithms are relevant to both IPv4 and IPv6 because they do not presuppose any particular network-layer protocol. Finally, we provide concluding rcmarks. Mulficast Routing Algorifhms Figure 3 shows an undirected graph G = (V, E), where Vis the set of nodes and E the set of links. Note that, siuce graph G is undirected, it models a communication network which has symmetric links. Let M = (CA1, TX, IL, NY) be a multicast group. (Shaded nodes in Fig. 3 belong to the multicast group.) Now, in order to perform multicast communication, the nodes in the multicast group must he interconnected by a tree. Thus, the problem of multicast routing in communication networks is equivalent to finding a tree Tin graph G such that T spans all vertices in the multicast group M. Such a tree is called a multicast lree and is shown in Fig. 3 by thick lines.* (The term Steiner free used in Fig. 3 will be clarified next.) Just as multicast communication can be of two types, multicast trees can also be classified into two corresponding categories: source-specific (or source-rooted) and group-shared. For the same multicast example as in Fig. 3, Fig. 4a shows a sourcespecific multicast tree which employs unidirectional links? (with source = CAl), while Fig. 4b shows a group-shared multicast tree. The key difference between a source-specific multicast tree and a group-shared multicast tree is that a source-specific multicast tree is optimized for source-specific multicast communication, while a group-shared multicast tree is optimized for group-shared multicast communication. For example, if we want to minimize the average delay for sourcespecific communication, we need to minimize the average sourcc-specific delay which is calculated by taking thc avcragc of the end-to-end delays over all (source, multicast-member) pairs. Now, assuming that each link in Fig. 4a has delay equal to 1, the source-specific delay of thc source-specific tree rooted Throughout thir work, the default weight of all links, unless pecrfied othenvise, is equal to 1. Note that a source-specific multicast free cormecls n source node to other nodes in the mellicast group by employing either unidi~ctional or bidirectional links, while a pup-shared mullicast tree employs on& bidirectional link. at CA1 is equal to 2.33 (the average of the delay from source CA1 to nodes TX, IL, and Ny). In comparison, the source-specific delay (with CA1 as the source) of thc group-shared multicast tree shown in Fig. 4b is equal to 3.33. On the other hand, if we were to calculate the average group-shared dclay of the source-specific tree by taking the average of the end-to-end delays over all (multicast-member. multicast-member) pairs, the average group-shared delay is equal to 3.5 in Fig. 4a, while the average group-shared delay of the group-shared tree in Fig. 4h is equal to 2.67. Thus, the application requirements dictate which type of multicast trees are “better.” The following is a list of the properties of a good multicast tree. Since for most multicast applications some properties are more important than others, we have divided the properties into thrce priority levels:4 high, medium, and low. High Priority * Low cost: Thc cost (or weight) of a multicast tree is the sum of the costs (or weights) of all the links in the multicast trcc. A good multicast tree tries to minimize this cost. - Low delay: Thc end-to-end delay from the source node to the destinatioii node is thc sum of the individual link delays along the route. A good multicast tree tries to minimize the end-to-end delay for every sourcc-dcstioation pair in the multicast group. Scalability: A good multicast tree is scalable in two respects. First, constructing a multicast trec for a large multicast group should requirc reasonable amounts of time and resources. Second, the switches in the communication network should he able to simultaneously support a large numbcr of multicast trees. Medium Priority Support for dynamic multicast groups: Multicast groups can he classified as static and dynamic. The members of a static multicast group do not change over time; in a dynamic multicast group, new membcrs may join or existing members leave. A good multicast tree should allow multicast members to join or leave the multicast trcc in a seamless fashion. Moreover, the properties of a good multicast tree should not degrade due to the dynamic nature of thc multicast grnup. Survivability: A good multicast tree should bc ablc to survive multiplc node and link failures. Note that the priori& 1eveI.v may he different for certain applicalions. For erample, while the jkimempmpeny ofn mullicast tree i.v not wry impor; tant in general, it may be the mmt impnrtamproper& of the midticast tree if the multicust tree is being employed by n multiplaycrgamr. Moreover; although Some properties arc runsidered low-priority for today’s applications, they may become more implant in thefite,a due to enzerRi,tg applications which may he beyond our cumprekension today. 92 IEEE Network * JanuarylFebruary 2000 low Priorify * Fairness: A good multicast tree is fair in two respects. First, it tries to provide a minimum quality of service (e.g., hounded delay) to each member in the multicast group. (It is not fair to unncccssarily punish one membcr in order to improve thc quality of service to other mcmbers.) Second, it tries to evenly dividc the multicasting effort (c.g., packet duplication effort) among the participating nodes. Most algorithms that havc been proposed in the literature mainly focus 011 Cost and delay OPtimization, although the nther Properties havc also heen addressed to a lesser extent. Before we examine each of the above prnpcrties in detail, let us examine some important theoretical concepts and dcfinitions which will help us better undcrstand the nature of the multicast routing problem. W Figure 5. An mmple of agraph for which the Steiner tree can be found by employingthe reduction rules. Cost of all Iinh = 1; costofthe Steinertree = 4. j) can he removed from G. Furthermore, if cji = dy and there is a path of cost dij from i to j not containing (i, j), then link (i, j) can be removed from G. 4)If G contains thrce distinct nodes U, v, w E M, such that U and v arc adjaccnt, cuy > d,,, and cuv > d,,, then link (U, v) can bc removed from G. In other words, if U, v, and w are any three nodes in the multicast group such that the cost of the link (U, v) is more than the cost of a path from nodc w to node U as well as node v, then link (U, v) does not belong 5)Let u E M. Let v and w be the closest and second closest adjacent nodes to U, respcctively. Now, if cBV t min{d, Ip E M and p # U} S c,,,, then the link (U, v) belongs to the Stciner tree and G can he contracted along (U, v). In other words, if the closest adjaccnt node (v) brings you nearer to other members of the multicast group, then link (U, v) should belong to the Steiner tree. For example, the graph shown in Fig. 5 (nodes in the multicast set are shaded) can he reduced to a single node by cmploying reduction 5 repetitively as follows. First, we contract along link (CA2, CAI); second, we contract along link (MI, NY); third, wc contract along link (C41, UT); and finally, we contract along link (UT, MO. Unfortunately, as the following lemma demonstrates, these reductions cannot he applied to a large number of instances of SPN which occur in typical communication networks. Usually, these reductions cannot he applied to cases in which IMI << I VI, G is not sparse: and G satisfies the triangle inequality (to he explained shortly). The following Lemma describes a sufficient condition for an instance of SPN to he “irreducible.” Lemma 1 - If an instance of SPN (say P) satisfies all of the following three conditions, then P cannot he reduced to a smaller instancc of SPN by using the aforementioned reduction rules. 1.The graph satisfies the triangle inequality, that is, the cost cUy of a link (U, v) is strictly less than the cost of any path from node u to nodc v which does not include link (U, v). 2.Thc minimum degree of the graph is 3, that is, Vv E V, dcg(v) 2 3. 3.Nonc of the nodes in the multicast group are adjacent to one another, that is, Vu, v t M, (U, v) e E. Thc classical optimization problem in multicast routing is called the Steiner tree problem in networks (SPN), and is defined as follows. Given An undirectcd graph G = (V, E) A cost function which assigns a positive real cost cZ,” to link v) to thc Steiner tree. - A set of nodes M L Vwhich belong to the multicast group find a tree T = (V, Er) which spans M, such that its cost CT = E<, dg, where djj is the cost of the shortest-path between nodcs i and j, then link (i, Proof - Reductions 1 and 2 cannot he applied because the minimum degree of the graph is 3. Rcdnction 3 cannot be applied because the graph satisfies the triangle inequality. Reduction 4 cannot he applicd hccause none of the nodes in thc multicast group are adjacent to one another. Finally, reduction 5 cannot bc applied because the graph satisfies the triangle * There are ,come other special caSes of SPN for which polynomial-time algorithms exist [l]. We define graph G lo be sparse ifall the spanning trees ofgraph G can be enumerated in polynomial fime. IEEE Network JanualyiFcbrualy 2000 93 inequality, and none of the nodes in the multicast group are adjacent to one another. The graph shown in Fig. 3 satisfies all w Thus, for typical wmmunication networks, it may he impossible to find a Steiner tree in a reasonable amount of time; hence, it is important to develop apprmimation algorithms for SPN. Approximation algorithms for SPN run in polynomial time and produce good-quality (hut not necessarily optimal) solutions to SPN. For some approximation algorithms, it is possible to prove aperformanceguarantee (i.e., a bound on the quality of the solution). A formal definition of performancc guarantee is as follows. Let r he a class of problems (such as SPN) and P E r he a problem instance. Let A(P) denote the cost of the solution found by algorithmA and OPT(P) denotc the cost of the optimal solution. We define the performance guarantee of algorithmA as II, = maxp,riA(P)lOPT(P)}. In other words, if the performance guarantee of an algorithm is cqual to p, then for all problem instances P E r, thc approximate solution is guaranteed to he at most p times costlier than the optimal solution. While most approximation algorithms for SPN have a pcrformance guarantee of 2, to the bcst of our knowledge, none of the known approximation algorithms have a performance guarantee better than 1116 [12]. In the following subsections, we examine the six properties of a multicast tree that were mentioned at the beginning of this section, paying more attention to the higher-priority properties, particularly cost and delay. Cost Optimization Approximation algorithms for optimizing the cost of a multicast tree employ different kinds of heuristics [13-161. Recall that if the multicast group consists of all the nodes in the graph, the problem reduces to the well-known minimum spanning tree problem. Thus, it is no surprise that some approximation algorithms are based on the so-called minimum spanning tree heuristic. One such approximation algorithm which was proposed by Kou, Markowsky, and Berman (henceforth referred to as KMB) [13] is examined below. KMB consists of five steps. First, using the nodcs in thc multicast group, we construct an undirected closure graph GI; thus, for every node pair (U, v) in the multicast group M, GI has an edge (U. v), such that the weight of the edge (c’.,,) is equal to the weight of the shortest path (d,

6. How routing table is constructed in fisheye state routing protocol? Explain in detail. [CO3-H1]

Based on the routing information update mechanism

Ad hoc wireless network routing protocols can be classified into 3 major categories based on the routing information update mechanism. They are:

Proactive or table-driven routing protocols:

- Every node maintains the network topology information in the form of routing tables byperiodically exchanging routing information.
- Routing information is generally flooded in the whole network.
- Whenever a node requires a path to a destination, it runs an appropriate path-finding algorithm on the topology information it maintains.

***Reactive or on-demand routing protocols*:**

- Do not maintain the network topology information.
- Obtain the necessary path when it is required, by using a connection establishment process.

Hybrid routing protocols:

- Combine the best features of the above two categories.
- Nodes within a certain distance from the node concerned, or within a particular geographical region, are said to be within the routing zone of the given node.
- For routing within this zone, a table-driven approach is used.
- For nodes that are located beyond this zone, an on-demand approach is used.

TABLE-DRIVEN ROUTING PROTOCOLS

- These protocols are extensions of the wired network routing protocols
- They maintain the global topology information in the form of tables at every node
- Tables are updated frequently in order to maintain consistent and accurate network state information
- Ex: Destination sequenced distance vector routing protocol (DSDV), wireless routing protocol (WRP), source-tree adaptive routing protocol (STAR) and cluster-head gateway switch routing protocol (CGSR).

Destination sequenced distance-vector routing protocol

- It is an enhanced version of the distributed Bellman -Ford algorithm where each node maintains a table that contains the shortest distance and the first node on the shortest path to every other node in the network.
- It incorporates table updates with increasing sequence number tags to prevent loops, to counter thecount-to-infinity problem, and for faster convergence.
- As it is a table-driven routing protocol, routes to all destinations are readily available at every node at all times.
- The tables are exchanged between neighbors at regular intervals to keep an up -to-date view of the network topology.
- The table updates are of two types:

Incremental updates: Takes a single network data packet unit (NDPU). These are used when a node does not observe significant changes in the local topology.
Full dumps: Takes multiple NDPUs. It is done either when the local topology changes significantly or when an incremental update requires more than a single NDPU.

- Table updates are initiated by a destination with a new sequence number which is always greater than the previous one.
- Consider the example as shown in figure (a). Here node 1 is the source node and node 15 is the destination. As all the nodes maintain global topology information, the route is already available as shown in figure (b).
- Here the routing table node 1 indicates that the shortest route to the destination node is available through node 5 and the distance to it is 4 hops, as depicted in figure (b)
- The reconfiguration of a path used by an on-going data transfer session is handled by the protocol in the following way.
- The end node of the broken link initiates a table update message with the broken link's weight assigned to infinity (∞) and with a sequence number greater than the stored sequence number for that destination.
- Each node upon receiving an update with weight ∞, quickly disseminates it to its neighbors in order to propagate the broken-link information to the whole network.
- A node always assign an odd number to the link break update to differentiate it from the even sequence number generated by the destination.
- Figure 7.6 shows the case when node 11 moves from its current position.

. **Advantages**

- Less delay involved in the route setup process.
- Mechanism of incremental update with sequence number tags makes the existing wired network protocols adaptable to ad hoc wireless networks.

- The updates are propagated throughout the network in order to maintain an up-to-date view of the network topology at all nodes.

Disadvantages

- The updates due to broken links lead to a heavy control overhead during high mobility.
- Even a small network with high mobility or a large network with low mobility can completely choke the available bandwidth.
- Suffers from excessive control overhead.
- In order to obtain information about a particular destination node, a node has to wait for a table update message initiated by the same destination node.
- This delay could result in state routing information at nodes.

Wireless Routing Protocol (WRP)

- WRP is similar to DSDV; it inherits the properties of the distributed bellman-ford algorithm.
- To counter the count-to-infinity problem and to enable faster convergence, it employs a unique method of maintaining information regarding the shortest distance to every destination node in the network and penultimate hop node on the path to every destination node.
- Maintains an up-to-date view of the network, every node has a readily available route to every destination node in the network.
- It differs from DSDV in table maintenance and in the update procedures.
- While DSDV maintains only one topology table, WRP uses a set of tables to maintain more accurate information.
- The table that are maintained by a node are :

Distance table (DT): contains the network view of the neighbors of a node. It contains a matrix where each element contains the distance and the penultimate node reported by the neighbor
for a particular destination.
Routing table (RT): contains the up-to-date view of the network for all known destinations. It keeps the shortest distance, the predecessor/penultimate node, the successor node, and a flag indicating the status of the path. The path status may be a simplest (correct) path or a loop (error), or destination node not marked (null).
Link cost table (LCT): contains the cost of relaying messages through each link. The cost of broken link is ∞.it also contains the number of update periods passed since the last successful update was received from that link.
Message retransmission list (MRL): contains an entry for every update message that is to be retransmitted and maintains a counter for each entry.

- After receiving the update message, a node not only updates the distance for transmitted neighbors but also checks the other neighbors' distance, hence convergence is much faster than DSDV.
- Consider the example shown in figure below, where the source of the route is node 1 and destination is node 15. As WRP proactively maintains the route to all destinations, the route to any destination node is readily available at the source node.
- From the routing table shown, the route from node 1 to node 15 has the next node as node 2. The predecessor node of 15 corresponding to this route is route 12.
- The predecessor information helps WRP to converge quickly during link breaks.
- When a node detects a link break, it sends an update message to its neighbors with the link cost of the broken link set to ∞. After receiving the update message; all affected nodes update their minimum distances to the corresponding nodes. The node that initiated the update message then finds an alternative route, if available from its DT. Figure 7.8 shows route maintenance in WRP.

Unit – IV

WIRELESS SENSOR NETWORKS (WSNS) AND MAC PROTOCOLS

Part – A

1. What is wireless sensor network? [CO4-L1]

Wireless Sensor Networks (WSN) [Estrin 1999, Kahn 1999] are a recent application of ad hoc networks that is expected to find increasing deployment in coming years, as they enable reliable monitoring and analysis of unknown and untested environments. These networks are "data centric" i e., unlike traditional networks where data is requested from a specific node, data is requested based on certain attributes such as, "which area has temperature 100ºF".

2. What are the components of WSN? [CO4-L2]

A wireless sensor network (WSN) is a hardware and Software package that typically consists of four parts

a) 'Sensors' Connected to each node by a wired connection. In Our case, we use sensors that can measure soil moisture, electrical conductivity, soil temperature, water pressure, flow rate, or a range of weather variables (light, Air temperature, wind, humidity, etc.).

3. Write short notes on memory devices in WSN [CO4-L2]

Simple embedded microcontrollers, such as the Atmel or the Texas Instruments MSP 430. A decisive characteristic here is, apart from the critical power consumption, an answer to the important question whether and how these microcontrollers can be put into various operational and sleep modes, how many of these sleep modes exist, how long it takes and how much energy it costs to switch between these modes. Also, the required chip size and computational power and on-chip memory are important

4. Define: transceivers in WSN [CO4-L1]

Currently used radio transceivers include the RFM TR1001 or Infineon or Chip on devices; similar radio modems are available from various manufacturers. Typically, ASK or FSK is used, while the Berkeley Pico Nodes employ OOK modulation. Radio concepts like ultra-wideband are in an advanced stage (e.g., the projects undertaken by the IEEE 802.15 working group).

5. Define: noise figure [CO4-L2]

A node in a simulator acts as a software execution platform, a sensor host, as well as a communication terminal. In order for designers to focus on the application-level code, a node model typically provides or simulates a communication protocol stack, sensor behaviors (e.g., sensing noise),

6. Write short note on different operational states of transceiver in WSN [CO4-L2]

A crucial step forward would be the introduction of a reasonably working wake-up radio concept, which could either wake up all SNs in the vicinity of a sender or even only some directly addressed nodes. A wake-up radio allows a SN to sleep and to be wakened up by suitable transmissions from other nodes, using only a low-power detection circuit. Transmission media other than radio communication are also considered, e.g., optical communication or ultra-sound for underwaterapplications. However, this largely depends on the application

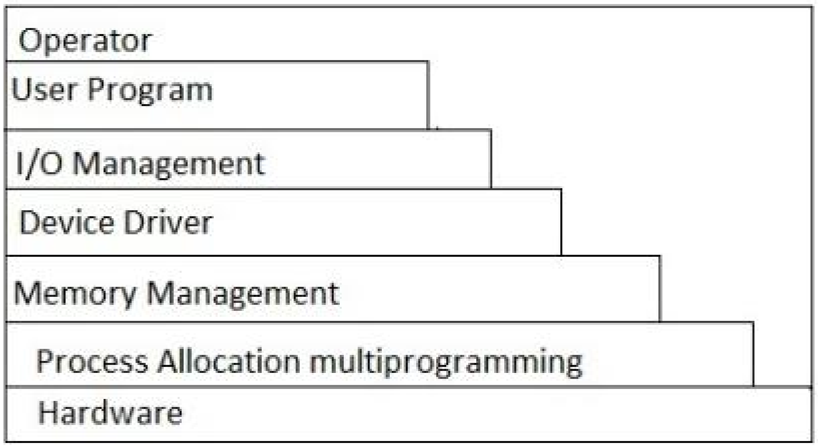
7. What are wakeup receivers? [CO4-L2]

A wake-up receiver is a low power radio-triggered device used to continuously monitor a channel and which activates the node on demand for incoming communication. Such a modular, small sized and good performing (-15 dBm input sensitivity and only 6µW power consumption) wake-up receiver module has been developed from off-theshelves components. A new software communication protocol taking advantage of the wake-up on radio functionality is proposed, successfully demonstrated and compares well against standard low duty cycle MAC protocols.

8. Write short notes on traditional concurrency process [CO4-L1]

A work methodology based on the parallelization of tasks (i.e. performing tasks concurrently), which is sometimes called Simultaneous Engineering or Integrated Product Development (IPD). It refers to an approach used in product development in which functions of design engineering, manufacturing engineering and other functions are integrated to reduce the elapsed time required to bring a new product to the market

9. Draw the structure of layered architecture **[CO4-L1]**



10. Define: LEACH **[CO4-L1]**

It is introduced in [Heinzelman 2000b] a hierarchical clustering algorithm for sensor networks, called LEACH (Low-Energy Adaptive Clustering Hierarchy). LEACH is actually a family of protocols [Heinzelman 2000b] which suggests two schemes, distributed and centralized, that have minimal setup time and are also very energy efficient

11. What is the MAC protocols used in sensor network? **[CO4-L2]**

Sensor nodes in a directed diffusion-based network are application aware, which enables diffusion to achieve energy savings by choosing empirically good paths and by caching and processing data in the network. An application of directed diffusion is to spontaneously propagate an important event to regions of the sensor network. Such type of information retrieval is well suited for persistent queries where requesting nodes expect data that satisfy a query for a period of time.

Part – B

1. Explain about the hardware components of sensor nodes **[CO4-L2]**

There are two types of programming for sensor networks, those carried out by end users and those performed by application developers. An end user may view a sensor network as a pool of data and interact with the network via queries. Just as with query languages for database systems like SQL, a good sensor network programming language should be expressive enough to encode application logic at a high level of abstraction, and at the same time be structured enough to allow efficient execution on the distributed platform. On the other hand, an application developer must provide end users a sensor network with the capabilities of data acquisition, processing, and storage. Unlike general distributed or database systems, collaborative signal and information processing (CSIP) software comprise reactive, concurrent, distributed programs running on ad hoc resource- constrained, unreliable computation and communication platforms. For example, signals are noisy, events can happen at the same time, communication and computation take time, communications may be unreliable, battery life is limited, and so on.

Sensor Node Hardware

Sensor node hardware can be grouped into three categories, each of which entails a different trade-offs in the design choices.

□Augmented general-purpose computers: Examples include low-power PCs, embedded PCs (e.g.PC104), custom-designed PCs, (e.g. Sensoria WINS NG nodes), and various personal digitalassistants (PDA). These nodes typically run –ff-the-shelf operating systems such as WinCE, Linux,or real-time operating systems and use standard wireless communication protocols such as IEEE802.11, Bluetooth, Zigbee etc. Because of their relatively higher processing capability, they canaccommodate wide variety of sensors, ranging from simple microphones to more sophisticated video cameras.

□Dedicated embedded sensor nodes: Examples include the Berkeley mote family [1], the UCLAMedusa family [2], Ember nodes and MIT □AMP [3]. T use commercialoff-the-shelf (COTS) chip sets with emphasis on small form factor, low power processing andcommunication, and simple sensor interfaces. Because of their COTS CPU, these platforms typicallysupport at least one programming language, such as C. However, in order to keep the programfootprint small to accommodate their small memory size, programmers of these platforms are givenfull access to hardware but rarely any operating system support. A classical example is the TinyOSplatform and its companion programming language, nesC.

□System on-chip (SoC) nodes: Examples of SoC hardware include smart dust [4], the BWRCpicoradio node [5], and the PASTA node [6]. Designers of these platforms try to push the hardwarelimits by fundamentally rethinking the hardware architecture trade-offs for a sensor node at the chipdesign level. The goal is to find new ways of integrating CMOS, MEMS, and RF technologies tobuild extremely low power and small footprint sensor nodes that still provide certain sensing,computation, and communication capabilities. Among these hardware platforms, the Berkeleymotes, due to their small form factor, open source software development, and commercialavailability, have gained wide popularity in the sensor network research.

Sensor Network Programming Challenges

Traditional programming technologies rely on operating systems to provide abstraction for processing, I/O, networking, and user interaction hardware. When applying such a model to programming networked embedded systems, such as sensor networks, the application programmers need to explicitly deal with message passing, event synchronization, interrupt handling, and sensor reading. As a result, an application is typically implemented as a finite state machine (FSM) that covers all extreme cases: unreliable communication channels, long delays, irregular arrival of messages, simultaneous events etc.

For resource-constrained embedded systems with real-time requirements, several mechanisms are used in embedded operating systems to reduce code size, improve response time, and reduce energy consumption. Microkernel technologies [7] modularize the operating system so that only the necessary parts are deployed with the application. Real-time scheduling [8] allocates resources to more urgent tasks so that they can be finished early. Event-driven execution allows the system to fall into low-power sleep mode when no interesting events need to be processed. At the extreme, embedded operating systems tend to expose more hardware controls to the programmers, who now have to directly face device drivers and scheduling algorithms, and optimize code at the assembly level. Although these techniques may work well for small, stand-alone embedded systems, they do not scale up for the programming of sensor networks for two reasons:

□ Sensor networks are large-scale distributed systems, where global properties are derivable from program execution in a massive number of distributed nodes. Distributed algorithms themselves are hard to implement, especially when infrastructure support is limited due to the ad hoc formation of the system and constrained power, memory, and bandwidth resources.

□ As sensor nodes deeply embed into the physical world, a sensor network should be able to respond to multiple concurrent stimuli at the speed of changes of the physical phenomena of interest.

There no single universal design methodology for all applications. Depending on the specific tasks of a sensor network and the way the sensor nodes are organized, certain methodologies and platforms may be better choices than others. For example, if the network is used for monitoring a small set of phenomena and the sensor nodes are organized in a simple star topology, then a client-server software model would be sufficient. If the network is used for monitoring a large area from a single access point (i.e., the base station), and if user queries can be decoupled into aggregations of sensor readings from a subset of nodes, then a tree structure that is rooted at the base station is a better choice. However, if the phenomena to be monitored are moving targets, as in the target tracking, then neither the simple client-server model nor the tree organization is optimal. More sophisticated design and methodologies and platforms are required.

Node-Level Software Platforms

Most design methodologies for sensor network software are node-centric, where programmers think in terms of how a node should behave in the environment. A node-level platform can be node-centric operating system, which provides hardware and networking abstractions of a sensor node to programmers, or it can be a language platform, which provides a library of components to programmers.

A typical operating system abstracts the hardware platform by providing a set of services for applications, including file management, memory allocation, task scheduling, peripheral device drivers, and networking. For embedded systems, due to their highly specialized applications and limited resources, their operating systems make different trade-offs when providing these services. For example, if there is no file management requirement, then a file system is obviously not needed. If there is no dynamic memory allocation, then memory management can be simplified. If prioritization among tasks is critical, then a more elaborate priority scheduling mechanism may be added.

Operating System: TinyOS

Tiny OS aims at supporting sensor network applications on resource-constrained hardware platforms, such as the Berkeley motes.

To ensure that an application code has an extremely small foot-print, TinyOS chooses to have no file system, supports only static memory allocation, implements a simple task model, and provides minimal device and networking abstractions. Furthermore, it takes a language-based application development approach so that only the necessary parts of the operating system are compiled with the application. To a certain extent, each TinyOS application is built into the operating system.

Like many operating systems, TinyOS organizes components into layers. Intuitively, the lower a layer is, the ‘closer’ it is to the hardware; the higher a layer is, the closer it is to the application. In addition to the layers, TinyOS has a unique component architecture and provides as a library a set of system software components. A components specification is independent of the components implementation. Although most components encapsulate software functionalities, some are just thin wrappers around hardware. An application, typically developed in the nesC language, wires these components together with other application-specific components.

A program executed in TinyOS has two contexts, tasks and events, which provide two sources of concurrency. Tasks are created (also called posted) by components to a task scheduler. The default implementation of the TinyOS scheduler maintains a task queue and invokes tasks according to the order in which they were posted. Thus tasks are deferred computation mechanisms. Tasks always run to completion without preempting or being preempted by other tasks. Thus tasks are non-preemptive. The scheduler invokes a new task from the task queue only when the current task has completed. When no tasks are available in the task queue, the scheduler puts the CPU into the sleep mode to save energy.

The ultimate sources of triggered execution are events from hardware: clock, digital inputs, or other kinds of interrupts. The execution of an interrupt handler is called an event context. The processing of events also runs to completion, but it preempts tasks and can be preempted by other events. Because there is no preemption mechanism among tasks and because events always preempt tasks, programmers are required to chop their code, especially the code in the event contexts, into small execution pieces, so that it will not block other tasks for too long.

Another trade-off between non-preemptive task execution and program reactiveness is the design of split-phase operations in TinyOS. Similar to the notion of asynchronous method calls in distributed computing, a split-phase operation separates the initiation of a method call from the return of the call. A call to split-phase operation returns immediately, without actually performing the body of the operation. The true execution of the operation is scheduled later; when the execution of the body finishes, the operation notifies the original caller through a separate method call.

In summary, many design decisions in TinyOS are made to ensure that it is extremely lightweight. Using a component architecture that contains all variables inside the components and disallowing dynamic memory allocation reduces the memory management overhead and makes the data memory usage statically analyzable. The simple concurrency model allows high concurrency with low thread maintenance overhead. However, the advantage of being lightweight is not without cost. Many hardware idiosyncrasies and complexities of concurrency management are left for the application programmers to handle. Several tools have been developed to give programmers language-level support for improving programming productivity and code robustness.

Imperative Language: nesC

nesC [9] is an extension of C to support and reflect the design of TinyOS. It provides a set of language constructs and restrictions to implement TinyOS components and applications.

A component in nesC has an interface specification and an implementation. To reflect the layered structure of TinyOS, interfaces of a nesC component are classified as *provides* or *uses* interfaces. A provides interface is a set of method calls exposed to the upper layers, while a uses interface is a set of method calls hiding the lower layer components. Methods in the interfaces can be grouped and named.

Although they have the same method call semantics, nesC distinguishes the directions of the interface calls between layers as *event* calls and *command* calls. An event call is a method call from a lower layer component to a higher layer component, while a command is the opposite.

The separation of interface type definitions from how they are used in the components promotes the reusability of standard interfaces. A component can provide and use the same interface type, so that it can act as a filter interposed between a client and a service. A component may even use or provide the same interface multiple times.

2. Explain about the software components of sensor nodes [CO4-L1]

Energy is of primary importance in wireless sensor networks. Sensor nodes have limited energy supplies and hence must make intelligent and informed decisions that can help conserve energy. Being able to make decisions based on knowledge of the current energy consumption can increase the lifetime with up to 52% [6]. Also, by distributing energy information to neighboring sensor nodes, routing can be made more energy-efficient [10, 13]. Current hardware platforms such as the Tmote Sky [7] and the ESB [9] do not provide hardware mechanisms for measuring the energy consumption of the sensor node. While it is possible to measure the battery voltage, the battery voltage is affected by the battery capture effect and is not a good estimate of the current energy consumption. Furthermore, the unique characteristics of sensor network applications make hardware-based energy measurement difficult [3]. In this paper we investigate the use of a software-based on-line energy estimation mechanism for small sensor nodes. The mechanism runs directly on the sensor nodes and provides real-time estimates of the current energy consumption. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. EmNets ’07, June 25-26, 2007, Cork, Ireland Copyright 2007 ACM 978-1-59593-694-3/07/06 ...$5.00. Our target platforms are low-end sensor nodes, such as the ESB [9] and the Tmote Sky [7]. The mechanism uses an intentionally simple linear model that is easy to implement and add to existing sensor node operating systems. No modifications to existing applications or network protocols are required. The idea of doing software-based on-line energy estimation is not new; for example, Younis and Fahmy have previously suggested the use of a simple linear model for estimating energy consumption [12]. However, their model requires extensive changes to all applications and protocols that use it. We are not aware of any previous work that evaluates the efficiency of software-based on-line energy estimation. The contribution of this paper is that we show that an intentionally simple mechanism for on-line node-level energy estimation can provide a good estimation of energy consumption. We have implemented our mechanism in Contiki [2] but the mechanism is general enough to be used in any operating system for sensor nodes. We evaluate the accuracy of the energy estimates by comparing the results from the energy estimation mechanism with the lifetime of sensor nodes. Our results show that the energy estimation provides good estimates, but further study is needed to quantify the accuracy of the mechanism. Hardware-based energy measurement mechanisms are typically difficult to add to existing hardware designs as they require a significant amount of modifications [3]. The cost increase for adding energy measurement is expected to be 100% [3] In contrast, our software-based energy estimation mechanism is easily added to existing hardware and software designs, without any additional per-unit cost. In this paper we show that a software-based energy estimation mechanism can be added to an existing sensor node operating system by adding only a few lines of code in specific operating system functions. No changes are needed to the applications or network protocols. The rest of this paper is structured as follows. We present our energy estimation mechanism in Section 2 and its implementation in Contiki in Section 3. We discuss calibration in Section 4. We evaluate the mechanism in Section 5 and review related work in Section 6. We discuss future work in Section 7 and conclude the paper in Section 8. 2. ON-LINE NODE-LEVEL ENERGY ESTIMATION To conserve energy, sensor nodes frequently switch on and off their components, such as the communication device, LEDs, or sensors. Our energy estimation mechanism is in- 1 2 3 4 5 6 7 8 9 10 11 1 1.5 2 2.5 3 3.5 4 Current (mA) Time (s) Measured current 0 10 20 30 40 50 60 70 1 1.5 2 2.5 3 3.5 4 Energy (mJ) Time (s) Energy 0 10 20 30 40 50 60 70 1 1.5 2 2.5 3 3.5 4 Energy (mJ) Time (s) Estimated energy Figure 1: Left: measured current draw. Middle: energy consumption over time, calculated by integrating the left figure. Right: estimated energy consumption over time, obtained from the on-line energy estimation mechanism. voked every time a hardware component is switched on or off. When a component is switched on, the estimation mechanism stores a time stamp. As the component is switched off again, a time difference is produced and added to the total time that the component has been turned on. The estimation mechanism keeps a list of all components and the time of which they have been turned on. The mechanism then uses the current draw of each component to produce an estimate of the total energy consumption. 2.1 Example Figure 1 shows the operation of the energy estimation mechanism. The left graph shows the measured current draw for a part of a typical sensing application. The application turns on the on-board temperature sensor and reads its value (1.5s - 2.5s). It then turns on the radio for one second to listen for any incoming traffic before it sends its sensor reading to its neighbor (2.5s - 3.5s). After sending its value, it signals that it is alive by blinking the green LED for half a second (3.5s - 4s). The middle graph shows the energy consumption and is obtained by integration of the left graph. The right graph in Figure 1 shows the estimated energy consumption of the above activity, as obtained from our on-line energy estimation mechanism. As the mechanism estimates the energy consumption only when a component is turned off, the graph shows jumps when the components are powered down. The energy consumption of the CPU is estimated every time the processor is switched from low power mode to normal mode, and therefore shows up as an increasing line. The saw tooth-pattern in the right graph is due to the intervals at which the energy estimate was sampled. 2.2 Energy Model The on-line energy estimation mechanism uses a linear model for the sensor node energy consumption. The total energy consumption E is defined as E V = Imtm + Iltl + Ittt + Irtr + X i Ici tci , (1) where V is the supply voltage, Im the current draw of the microprocessor when running, tm the time in which the microprocessor has been running, Il and tl the current draw and the time of the microprocessor in low power mode, It and tt the current draw and the time of the communication device in transmit mode, Ir and tr the current draw and time of the communication device in receive mode, and Ici and tci the current draw and time of other components such as sensors and LEDs. The energy model does not contain a term for the idle current draw of the board itself; this is embedded in the low power mode draw of the microprocessor. In many cases, the voltage V does not need to be explicitly computed, as the energy estimate often is used only for comparison between different nodes. If all nodes have the same voltage, computed E/V values can be compared directly, without the need of a multiplication operation. 3. IMPLEMENTATION The implementation of the on-line energy estimation mechanism requires only small changes to existing operating system source code. We have implemented the mechanism in the Contiki operating system [2] but the mechanism can easily be incorporated into other sensor node operating systems. The energy estimation module maintains a table with entries for all components, the CPU, and the radio transceiver. Each table entry contains the total time that the corresponding component has been turned on. Energy estimation is implemented in two lines of code in the device driver for the hardware for which energy is to be estimated. When the component is turned on, the energy estimation module is called to produce a time stamp. When the component is turned off, the time difference from when the component was turned on is computed. The time difference is added to the table entry for the component. To add energy estimation to an existing device driver, only two lines of code needs to be added. Time measurement is implemented using the on-chip timers of the MSP430. Since the on-chip timers work even when the microcontroller is in low-power mode, the time measurement is non-intrusive. We use the 32768 Hz clock divided by 8, producing 4096 clock ticks per second. 4. CALIBRATION To be able to accurately estimate the energy, the estimation mechanism must know the current draw for the microprocessor, communication device, and the components to be used in Equation 1. Calibration is performed by off-line measurement of the current draw for the sensor board using an oscilloscope. The average current draw for the different 0 200 400 600 800 1000 1200 1 2 3 Estimated E/V (mAs) Program Estimated energy Standard deviation Figure 2: The estimated energy in the last packet before node failure. Ideally, the estimated energy should be the same regardless of program. parts of the board is then used as input to the estimation mechanism. 5. EVALUATION To evaluate the efficiency of the on-line energy estimation mechanism, we use the method developed by Ritter el al. [8] where ESB sensor nodes [9] are equipped with 1F capacitors. The capacitors have a predictable energy storage and energy dissipation rate [11]. The capacitor is charged by turning on the power switch on the sensor node, which causes the battery to be connected to the capacitor. When the battery is switched off, the node runs on the energy contained in the capacitor. The energy stored in the capacitor can power the ESB node for a few minutes, depending on the energy consumption of the software running on the node. The program shown in Figure 1 runs for about four minutes from the energy in the capacitor. We use two different ESB nodes and run all experiments on both nodes. We run three different programs that emulate a standard sensor network application using the sensors, turning the radio and LEDs on and off at regular intervals, and sending data packets over the radio. The intervals for the three programs are configured to be different. In addition to turning components on and off, the sensor node transmits its estimated energy consumption via radio once every two seconds. The node runs until it dies. We define the lifetime of the node to be the time until the node sends its last packet. A base station node listens to the radio traffic and sends it via a serial cable to a PC, which logs the data to a file. 5.1 Estimation Accuracy To evaluate the energy estimation accuracy, we run the three programs on capacitor-equipped ESB nodes and capture the estimated energy that the nodes transmits over the radio. Ideally, the nodes should report the same energy estimate in the last packet they transmit before they die. The choice of program should not affect the estimated energy. Figure 2 shows the energy estimate in the last packet before node failure. The figure shows that the estimated energy varies both between programs and within each proTable 1: Code and memory footprint. Code Memory Module (bytes) (bytes) Book keeping 54 48 Summation 340 0 gram. This is due to the low precision of the experimental evaluation mechanism, which involves a fair amount of manual intervention. For example, we need to manually turn on and off the nodes to charge and discharge the capacitor. We plan to investigate validation mechanisms with higher precision [3] as part of future work. 5.2 Overhead of the Estimation Mechanism The energy estimation mechanism measures and stores the time during which hardware components have been turned on. The mechanism therefore incurs an overhead in terms of processing power. However, the code required to measure and store the time is very small; only 11 processor cycles to store a time stamp before turning on the hardware component and 20 processor cycles to update the total time after the component has been switched off. We measured the number of times the time stamping code was invoked in the data collection case study below and found that, on the average, the time stamping code was run 60 times per second. This incurs an overhead of approximately 1800 cycles per second, or 0.7% of the total processing time. The energy overhead of these few extra cycles is negligible. The code footprint and memory requirements of the mechanism are small, as shown in Table 1. The book keeping module keeps track of how long the components have been turned on. The summation module calculates Equation 1 and must be altered depending on the calibrated parameters for the particular hardware device on which the mechanism is executed. 5.3 Case Study: X-MAC As an example of how our energy estimation mechanism is intended to be used, we estimate the energy overhead of the X-MAC duty-cycling radio protocol [1] using the softwarebased on-line energy estimation mechanism. The X-MAC protocol switches on and off the radio at regular intervals to conserve the energy of the sensor node. When a node is to send a packet, it first broadcasts a train of short strobe packets. When the other nodes hear a strobe packet, it turns on its radio in preparation of receiving a full packet. As an optimization for unicast packets, the strobe packets include the address of the receiver of the full packet. When the receiver hears the strobe packet, it immediately sends a short acknowledgement packet to the sender of the strobe packets. The sender can then immediately send its full packet. All other nodes that overhear the packets can turn off their radios until the full packet has been transmitted. In this case study, we are interested in experimentally evaluating the unicast-packet optimization of X-MAC. We implement the X-MAC protocol in Contiki and setup a data collection network consisting of nine Tmote Sky nodes. One node acts as a base station; it collects the data from the network, and writes it to a PC. The other eight nodes are 0 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4 1 2 3 4 5 6 7 8 Normalized total power consumption Sensor node Radio, listening Radio, transmitting CPU, active CPU, idle 0 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4 1 2 3 4 5 6 7 8 Normalized total power consumption Sensor node Radio, listening Radio, transmitting CPU, active CPU, idle Figure 3: The estimated power consumption of eight nodes running the Contiki data collection protocol with X-MAC. The left graph is without the unicast optimization and the right graph with optimization. The power is normalized to the maximum Tmote Sky power consumption. arranged so that they form a two-hop network. The data collection protocol sends energy estimates produced by the energy estimation mechanism. We configure the X-MAC protocol to have a duty cycle of approximately 9%, based on the estimated number of packets per second and the analytical results calculated by the authors of X-MAC [1]. The estimated energy is shown in Figure 3. The left graph shows the estimated energy without the unicast optimization and the right graph the estimated energy with the optimization. We see that the optimization is able to obtain a significant reduction in energy consumption.

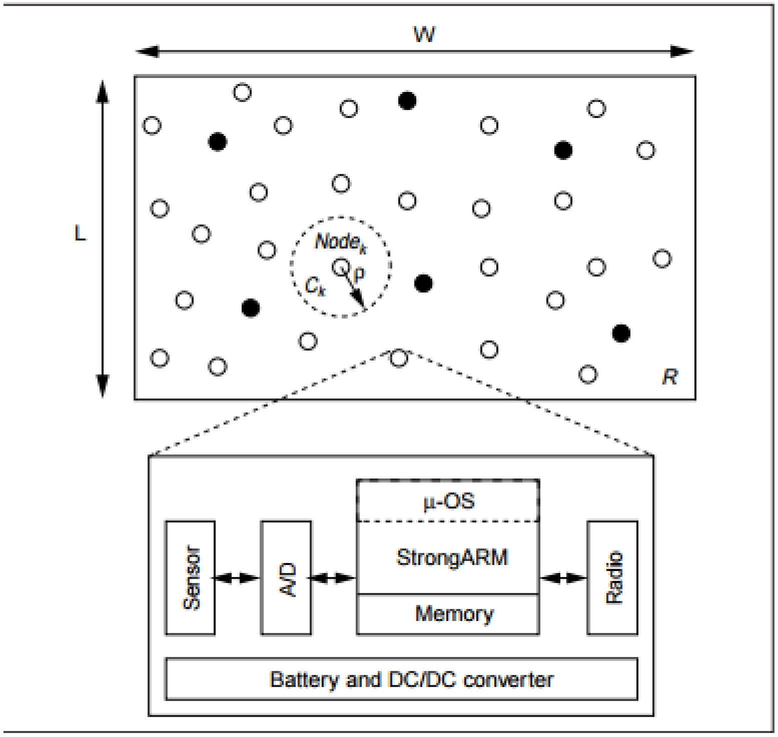
Furthermore, we see that idle listening is dominating the total energy consumption. Finally, we see that the energy consumption of nodes 2 and 5 is significantly higher than the energy consumption of the other nodes. This is because those nodes happened to route packets for the two-hop nodes in our experiment. This behavior would have been difficult to find if we would have measured the energy of a single node only. We take this as an indication that a systems perspective is useful when evaluating the energy consumption of sensor network protocols. 6. RELATED WORK Jiang et al. [3] show that the unique characteristics of sensor network applications make it difficult to measure the energy consumption of sensor nodes. The authors develop a hardware-based mechanism for measuring the energy consumption of sensor nodes that they expect to have a per-unit cost similar to that of the sensor node. It therefore incurs a significantly higher cost than software-based energy estimation. However, hardware-based mechanisms are able to capture phenomena such as per-node fluctuations in energy consumption that are not possible to study using our software-based mechanism. We expect both hardware-based and software-based methods to be used in the future. Operating systems for wireless sensor networks such as TinyOS, SOS, Mantis, and Contiki, reduce their energy consumption by powering off the microcontroller and hardware components when they are not used. Our work is orthogonal to this; on-line energy estimation estimates the actual energy consumption of the devices rather than trying to reduce the energy consumption. Younis and Fahmy [12] use a linear model for on-line estimation of node-level energy consumption, but do not evaluate the mechanism’s effectiveness. Furthermore, their model requires all applications to be explicitly rewritten to estimate their own energy consumption. In contrast, our model does not require any modifications to applications or network protocols. There are many sensor network simulators with energy estimation abilities [4, 5]. However, simulators are run off-line and cannot estimate the energy consumption in an on-line sensor network. Unlike off-line emulation, on-line energy estimation makes it possible to do energy-aware decisions about routing and transmission power scheduling, which potentially can prolong sensor node lifetime [6, 10, 13]. Landsiedel, Wehrle, and G¨otz [4] estimate the energy consumption of TinyOS-based systems using an off-line emulator. Our work is different in that it does on-line energy estimation. The off-line emulation must accurately capture the time in the processor is awake, and significant effort is devoted to doing accurate time emulation. With on-line energy estimation, time measurements can be directly obtained from on-chip timers provided by the microprocessor. Furthermore, the effects of interrupts and timers, that need to be carefully emulated in an off-line estimator, are automatically included in the energy estimates obtained with on-line estimation. The need for network-level energy monitoring in sensor networks is discussed by Zhao, Govindan, and Estrin [13]. Their work addresses the problem of transmitting node energy levels across the network. The monitoring mechanism, which is developed for a significantly larger target platform than ours, assumes the existence of a mechanism for measuring or estimating the node-level energy, such as ACPI. Our work addresses the problem of providing a way to estimate node-level energy, even for devices without ACPI, and is thus orthogonal to their work.

3. With a neat diagram, Explain the sensor network architecture [CO4-L2]

In this section, we address the key issues related to the architecture and synthesis of an individual SN node. Architectural aspects are discussed along three lines: hardware, software, and middleware. While design issues are presented from synthesis and analysis points of view. There have been at least three main approaches in which the architecture of SN nodes has been addressed [Reconsider wording]. The first group of initial efforts is a number of designs of individual sensor nodes and badges [Agr99, Asa98, Loc02, Mag98, Men01, Pot00, Wan92]. The emphasis of this class is ensuring the creation of working prototypes and, in some cases, pushing the state-of-the-art of an individual component (e.g. radio, low power, energy harvesting). The second group was represented by the Mote/TinyOS development team at UC Berkeley [Cul01, Hil00]. They made the first effort to address the trade-offs between various components of the node by developing new architectures and operating systems (OS). The main characteristic of the last group of efforts is sensor centered. The emphasis is to exploit relatively inexpensive off-the-shelf components in terms of cost and energy as a basis to explore qualitative and quantitative trade-offs between node components and in particularly sensors. It is difficult to anticipate technological trends, but one can easily identify at least some high impact trends and required solutions. For example, it is apparent that there is a need for overall energy consumption balanced architectures. Another high impact research topic is sensor organization and development of the interface between components. Finally, due to privacy, security and authentication needs, techniques such as unique ID for CPU and other components that facilitate privacy will be in high demand. In the software domain, the main emphasis will be on RTOS (Real Time Operating System) [Li97]. There is a need for ultra aggressive low power management due to energy constraints and a need for comprehensive resource accounting due to demands for privacy and security. In a number of cases support for mobility functions (e.g. location discovery) are also needed. Middleware will be in even stronger demand in order to enable rapid development and deployment of new applications. Tasks such as sensor data filtering, compression, sensor data fusion, sensor data searching and profiling, exposure coverage and tracking will be ubiquitous. Synthesis of sensor nodes will pose a number of new problems in the CAD world. It is obvious that new types of models, abstractions, and tasks will be defined and solved. For example, sensor allocation and selection, sensor positioning, sensor assignment, and efficient techniques for sensor data storage are typical examples of pending synthesis tasks. Development of conceptually simple, clean, and inexpressive models of computation is of prime importance as a starting point for synthesis of modern computing systems. The sensor nodes will require not just new models of computations, but also new models of the physical world. One such example is standard Euclidian space with classical physical laws (e.g. Newton's law, Thermodynamics law). It is well known that parts that are responsible for modern design flow, debugging and verification are the most expensive and time consuming components. Due to the heterogeneous nature and the complex interactions between components, we expect the same in the case of sensor nodes. In particular, we anticipate that the techniques for error and fault discovery, testing, and calibration will be of prime importance. In the rest of this section, we describe four representative SN nodes designs: Berkeley Mote, Piconode nodes, UCLA Medusa II and light compass node. 5.1 BERKELEY MOTE NODE The starting point for designing modern computer systems is a comprehensive set of benchmarks that are representative for common users. Unfortunately, such a set of benchmarks is not currently available to designers of SN nodes. The starting point for designing Mote wireless sensor network nodes was the set of qualitative observations about the requirements of wireless sensor networks. Special emphasis is placed on small physical size and low energy consumption. In addition, attempts have been made to facilitate concurrency intensive operations, in order to provide control hierarchy and take advantage of the limited physical parallelism. Furthermore, the design decisions are driven by retargetability for robust operations at least at the network level. The design went though several iterations and, until recently, was leveraging on the availability of standard off-the-shelf components. Generally speaking, the design is radio centric in the sense all main decisions are made is such a way to facilitate low energy communications. The main processor is Atmel 90LS8535 microcontroller that has 8-bit Harvard architecture with 16 bit addresses. It achieves speed of 4MHz at 3W. The system has rather minimal amount of memory that consists of 8 Kbytes of flash for program memory and 512 bytes SRAM for data memory. Therefore, the system can be integrated only with low frequency sampling sensors and it has to communicate frequently. The processor integrates a system of timers and counters and can be placed in four energy modes: active, idle, power down, and power save. In the idle mode, the processor is completely shut off. In the power down mode, only the watchdog and asynchronous interrupt logic is awake. Finally in the power save mode, the asynchronous timer is also active in addition to the watchdog and interrupt logic. The system also has co-processor Atmel 90LS2343 microcontroller that has 2 Kbytes flash instruction memory and 128 bytes of SRAM and EEPROM memory. The co-processor can be used to reprogram the main microcontroller. The authors consider the RF Monolithic 916.50 transceiver as the central part of the design. The radio is equipped with antenna and a system of discrete components that can be used to alter characteristics of the physical layer such as signal strength. The radio operates at speed of 19.2 Kbytes/sec. The transceiver can operate in three modes: transmission, reception and power off. The system can have up to 8 sensors. Two most widely used are photoelectric optical sensor and temperature sensor. Each sensor is placed on the bus that is controlled using software. It is instructive to consider the power characteristics of the design. The MCU core consumes between 2.5 to 6.5mA. The radio consumes between 5 to 12mA. The optical sensor and temperature sensor consume 0.3 to 1mA respectively. The coprocessor consumes 1 to 2.4mA. Finally, EEPROM consumes 1 to 3mA. In particular, it is instructive to compare energy spent for bit transmission and bit processing. The system spends about 1mJ to send and 0.5mJ to receive 1 bit. At the same time, the system can execute approximately 120 instructions for each mJ spent. The system does not have provision for energy reduction using variable voltage, therefore energy is saved mainly by turning the system off. The core of the system software for the design is an exceptionally compact microthreading operating system (TinyOS). The Berkeley design team concluded that new application domain required a new operating system, therefore they decided not to adopt any of a great variety of RTOS 8 bit controllers. While this decision certainly resulted in higher power efficiency and more interesting system software architecture, it also created additional demands and constraints in programming already highly constrained hardware. Nevertheless, the system has been highly popular in research community. Several thousands copies of various versions of the mote have been used by more than 200 research teams. The greatest strength in the system is its small size and low power. Probably the most serious disadvantages are related to the development of real applications. While motes have been tremendously popular in research community, it is still unclear how well they are suited for applications where more complex systems of sensors are needed. 5.2 UCLA MEDUSA MK-2 NODE The Medusa MK-2 node is representative of the state-of-the-art design of more powerful sensor nodes [Sav02]. The computational unit of Medusa MK-2 nodes consists of two microcontrollers. The first one is a 8-bit Atmel STMega128L MCU with 4MHz that has 32K of flash memory and 4KB of RAM. This processor serves as an interface between sensors and radio baseband processing. The second microcontroller is an ATMEL ARM THUMB processor enclosed within 120-ball BGA package. It has significantly more processing power at 40MHz. It also includes 136KB of RAM and 1MB of on-chip FLASH memory. The communication unit of Medusa MK-2 nodes is a combination of a TR 1000 low power radio from RF Monolithics for wireless and a RS-485 serial bus transceiver for wireline communication. The sensing unit has two components: a MEMS accelerometer and a temperature sensor. It can be also augmented with other types of sensors. Medusa nodes also incorporated a variety of interfaces, including 8 10-bit ADC inputs, serial ports and numerous general purpose I/O ports. An ultrasonic ranging unit is implemented on an accessory board that uses 40KHz transducers. Ultrasonic measurements are coordinated with RF measurements in order to calculate inter-node distances and therefore enable localization of nodes. Localization is conducted using iterative linearized multilateration. The nodes also have two external connectors. The first one is used to communicate with a PC to download and debug software. It also provides the necessary wiring requirements for connecting to an external GPS module. The second connector has a set of ADC and GPIO, and serves as an expansion slot for attaching add-on boards carrying different sensors. Finally, Medusa nodes also have two pushbuttons that serve as a user interface. They are mainlyused for triggering events and executing different tests during experimentations. It is interesting to take a closer look at the computational unit of Medusa Mk-2 nodes. According to the computation requirements, the computational tasks are classified into two board categories: low demand tasks and high demand low frequency processes. Low demanding tasks are the periodic processes such as baseband processing for the radio while listening for new packets, sensor samplings, handling of sensor events, and power management. Even though these tasks do usually require a high concurrency, they are not particularly demanding in terms of computational resource requirements and therefore can be easily handled by an 8-bit microcontroller. The Medusa-MK-2 nodes use a low power AVRMega128L microcontroller. The second category of low frequency high demanding tasks is related to the processing of acquired sensor data in order to produce user requested information. For example, in the case of the fine-grained localization problem, a sensor node is expected to compute an estimate of its own location based on a set of distance measurements to known beacons or neighbors. In order to avoid error propagation, a node has to perform a set of high precision operations. If an 8-bit processor is used to conduct this type of computations, it would result in high latencies and lower precision. Therefore a high-end processor is a more adequate solution. More specifically, Medusa adopts the 40MHz ARM THUMB processor to perform this type of operations. Another advantage is that the node can use existing standard applications and libraries. The THUMB microcontroller also has sufficient resources to support off the shelf embedded operating systems such as Red Hat, eCos, and uCLinux. The inclusion of the THUMB processor is also justified by a comparison of the two processors made from a power/latency perspective conducted by the UCLA group. The THUMB processor executes instructions at the rate of 0.9MIPS per MHz at 40Mhz while consuming 25mA with a 3V supply, which has a performance of 480 MIPS/Watt. On the other hand, the ATMega128L only provides 242 MIPS/Watt performance when operating at 4MHz and consumes 5mA at 3V supply. Communication between the two processors is handled by a pair of interrupt lines, one for each microcontroller, and a SPI bus. The two nodes remain in sleep mode until an interrupt indicating the need for data exchange occurs. The communication takes place over the 1Mbs SPI bus. Medusa MK-2 nodes are capable of two types of communications: wired and wireless. All nodes are equipped with both a wired link and a wireless link. The wireless link is a low power TR1000 radio from RF Monolithics. This radio has a maximum transmitting power of 0.75mW and has an approximate transmission range of 20 meters. Two modulation schemes are supported by this radio: On-Off Keying (OOK) and Amplitude Shift Keying (ASK). Selection of the appropriate modulation can be done in software. On a Medusa MK-2 node, the baseband processing for the radio is done by ATMega128L microcontroller. This also allows the node to run the low power S-MAC [Ye02] protocol on the ATMega128L processor. In addition to the wireless link, Medusa nodes also incorporate an RS-485 serial bus interface for wireline communication. By attaching a low power RS-485 transceiver to one of the RS-232 ports of the THUMB processor, it allows the node to connect to an RS-485 network using an RJ-11 connector and regular telephone wire. A single RS-485 has occupancy up to 32 nodes that span over a total wire length distance of 1000 feet. The power unit of Medusa MK-2 nodes consists of two main components: the power supply and the Power Management and Tracking Unit PMTU [Che02]. The power supply consists of a 540mAh lithium-ion rechargeable battery and an up-down DC-DC converter that has a 3.3V output that can reach [Generate?] up to 300mA of current from the battery. The power supply is designed in such a way that power-additional sensors can be attached later on as accessory boards, since the node only requires less than 50mA with no sensors attached. By putting the ARM THUMB processor together with the RS-485 and RS-232 transceivers in sleep mode most of the time, an 80% reduction of the overall node power consumption can be achieved in a typical sensor network setting. Comprehensive energy consumption comparisons between Medusa MK-2 nodes and other SN nodes designs can be found in [Sav02]. 5.3 BWRC PICONODE Another communication-centered sensor node design is the PicoNode [Rab00]. The main overall objective of this design is to simultaneously provide both flexibility and low energy consumption. It consists of four main modules. The first two units are processors: an embedded processor unit and configurable satellite units. The embedded processor is dedicated mainly for application and protocol-stack layers that require higher flexibility but have relatively low computational complexity and are infrequently requested. Configurable processing modules are targeted for the more frequent tasks with higher computational requirements. The other two modules are dedicated to communication tasks. They are a parameterized and configurable digital physical layer and a simple direct-down conversion RF front end. These modules are interconnected by a flexible and low-power consumption interconnect scheme. The authors claim that a dynamic matching between application and architecture leads to significant energy savings for signal- processing applications while maintaining implementation flexibility. One of the main premises of the design is the observation that the processor implementation is three orders of magnitude more expensive in terms of energy consumption than the implementations of the dedicated hardware. However, there is also the tradeoff between flexibility and programmability (software on programmable platforms) and energy consumption (ASIC hardware). The traditional approach is to design the wireless transceiver using only RF and analog circuit modules. More recently, a design approach that is mostly digitalized has emerged. This is inspired by the insight that digital circuits can improve exponentially with the scaling of technology, while analog circuits get linearly worse. This is mainly due to the reduction of the supply voltage. Therefore, it is beneficial to incorporate a small, noncritical analog front end and use digital back-end processing to balance the limitations. Many design challenges are related to the physical layer. They are mostly related to the low-energy targets and variable demands from the network. Therefore, in order to satisfy various demands from the network, the PicoNode physical layer is parameterizable. These parameters include power control modes, modulation scheme, and bit rate. In order to meet the low-energy requirement, the physical layer must meet two mutually exclusive criteria: fast signal acquisition and low standby power. The first criterion is referring to the process of waking up in the least amount of time, then receive bursts of data and immediately go back to sleeping mode after the data acquisition. The second criterion emphasizes on consuming the least amount of energy while sleeping. The reason why they are usually mutually exclusive is that there exists an inverse proportional relationship between the depth of sleep (i.e. energy consumed) during standby and the time required to wake up. PicoNode is designed in such a way that it does not require an interval power supply. It is self-constrained and selfpowered using the environment by harvesting energy from light and vibrations [Rab00]. There are two major constraints for harvesting ambient energy from the environment: applicability within the environment and the size of the node (Berkeley group targets the one-cubic-centimeter design). 5.4 SENSOR-CENTRIC DESISGN: LIGHT COMPASS The final sensor node design alternative that we will briefly overview is the light compass node [Won03]. The emphasis in this approach is completely shifted from computation, communication, and storage to sensors. The first three functions are provided by a standard laptop or PDA. The rationale is that this type of design will progress on its own to become a viable platform for sensor network nodes. Even the interface of the sensor is built using off-theshelf component. The focus is placed on sensors and how to select and place them in such a way that sensor data fusion is facilitated. In addition, special emphasis is place on how to rapidly develop retargetable sensor data fusion software and how to develop systematic procedures for design of sensor nodes. Figure 1 shows the light sensor components used. The smallest device (on the left) is a miniature silicon solar cell that is used for converting light impulses directly to electrical charges (photovoltaic). It generates its own power and therefore does not require any external bias. This silicon cell is further mounted on a 0.78cm x 0.58cm x 0.18cm thick plastic carrier that generates roughly 400 mV in moderate light (most typical rooms). A significantly larger sensor (on the right), referred to as a photoconductor, measures 2.54cm x 2.15cm and can be surrounded by a 0.18cm thick plastic encapsulated ceramic package. In strong light its resistance measures 20O, and 5KO in complete darkness. These components are very economically viable (roughly $0.30 each) and they can be easily purchased in large quantities. These sensors can be used in multiple prototypes such as the ones shown in figure 2. On the left of Figure 2, the sixsided cut-pyramid structure has base length of 3 cm and a top edge length of 1 cm with a 60o slope. Sensor(s) can be attached to each side of the structure depends on the application and purpose. The structure on the right is a cube with 2cm edges, therefore it can incorporate up to six sensors with one on each surface. In this light sensor platform, the heart component is an 8-channel analog to digital converter (ADC) module. It is used to read the sensor values through the parallel port of a standard PC laptop. This ADC component is comprised of a Maxim MAX186 ADC which has an internal analog multiplexer that can be configured for eight single-ended, or four differential inputs at a 12-bit resolution; with a conversion time that is under 10s. This component is pictured on the left in Figure 3. In addition, some of the other components of the circuit include: several resistors to protect the analog inputs; capacitors to filter noise; an external 4.096V voltage regulator; and an 8-bit digital latch required for parallel port communications. The overall design flow of a sensor appliance is presented in detail in Figure 4. The main goal of this design was to achieve low power consumption while maintain a tolerable level of coverage. Figure 5 – 7 depict the results obtained from four different sensor structures: a 4-sensor pyramid (square base), a 4- sensor cut pyramid (triangular based pyramid with a flat sensor on top), a 5-sensors pyramid (pentagonal base), and a 5-sensor cut pyramid (square base pyramid with a flat sensor on top). In all cases, the objective was to estimate the positions of 5000 randomly placed light instances.

4. Write notes on Dynamic Energy and power management **[CO4-L1]**

Wireless Distributed microsensor networks have gained importance in a wide spectrum of civil and military applications.1 Advances in MEMS (microelectromechanical systems) technology, combined with low-power, low-cost digital signal processors (DSPs) and radio frequency (RF) circuits have resulted in the feasibility of inexpensive and wireless microsensor networks. A distributed, self-configuring network of adaptive sensors has significant benefits. They can be used to remotely monitor inhospitable and toxic environments. A large class of benign environments also requires the deployment of a large number of sensors such as for intelligent patient monitoring, object tracking, and assembly line sensing. These networks' massively distributed nature provides increased resolution and fault tolerance compared to a single sensor node. Several projects that demonstrate the feasibility of sensor networks are underway.2 A wireless microsensor node is typically battery operated and therefore energy constrained.



To maximize the sensor node's lifetime after its deployment, other aspects—including circuits, architecture, algorithms, and protocols—have to be energy efficient. Once the system has been designed, additional energy savings can be attained by using dynamic power management (DMP) where the sensor node is shut down if no events occur.3 Such event-driven power consumption is critical to maximum battery life. In addition, the node should have a graceful energy-quality scalability so that the mission lifetime can be extended if the application demands, at the cost of sensing accuracy.4 Energy-scalable algorithms and protocols have been proposed for these energy-constrained situations. Sensing applications present a wide range of requirements in terms of data rates, computation, and average transmission distance. Protocols and algorithms have to be tuned for each application. Therefore embedded operating systems (OSs) and software will be critical for such microsensor networks because programmability will be a necessary requirement. We propose an OS-directed power management technique to improve the energy efficiency of sensor nodes. DPM is an effective tool in reducing system power consumption without significantly degrading performance. The basic idea is to shut down devices when not needed and wake them up when necessary. DPM, in general, is not a trivial problem. If the energy and performance overheads in sleepstate transition were negligible, then a simple greedy algorithm that makes the system enter the deepest sleep state when idling would be perfect. However, in reality, sleep-state transitioning has the overhead of storing processor state and turning off power. Waking up also takes a finite amount of time. Therefore, implementing the correct policy for sleep-state transitioning is critical for DPM success. While shutdown techniques can yield substantial energy savings in idle system states, additional energy savings are possible by optimizing the sensor node performance in the active state. Dynamic voltage scaling (DVS) is an effective technique for reducing CPU (central processing unit) energy.5 Most microprocessor systems are characterized by a time-varying computational load. Simply reducing the operating frequency during periods of reduced activity results in linear decreases in power consumption but does not affect the total energy consumed per task. Reducing the operating voltage implies greater critical path delays, which in turn compromises peak performance. Significant energy benefits can be achieved by recognizing that peak performance is not always required and therefore the processor's operating voltage and frequency can be dynamically adapted based on instantaneous processing requirement. The goal of DVS is to adapt the power supply and operating frequency to match the workload so the visible performance loss is negligible. The crux of the problem is that future workloads are often nondeterministic. The rate at which DVS is done also has a significant bearing on performance and energy. A low update rate implies greater workload averaging, which results in lower energy. The update energy and performance cost is also amortized over a longer time frame. On the other hand, a low update rate also implies a greater performance hit since the system will not respond to a sudden increase in workload. We propose a workload prediction strategy based on adaptive filtering of the past workload profile and analyze several filtering schemes. We also define a performance-hit metric, which we use to judge the efficacy of these schemes. Previous work evaluated some DVS algorithms on portable benchmarks.6 System models The following describes the models and policies, derived from actual hardware implementation. Sensor network and node model The fundamental idea in distributed-sensor applications is to incorporate sufficient processing power in each node so that they are self-configuring and adaptive. Figure 1 illustrates the basic sensor node architecture. Each node consists of the embedded sensor, analogdigital converter, a processor with memory (which, in our case, is the StrongARM SA-1100 processor), and the RF circuits. Each component is controlled by the microoperating system (µOS) through microdevice drivers. An important function of the µOS is to enable power management. Based on event statistics, the µOS decides which devices to turn off and on. Our network essentially consists of η homogeneous sensor nodes distributed over rectangular region ρ with dimensions W × L. Each node has visibility radius r. Three different communication models can be used for such a network: ■ direct transmission (every node transmits directly to the base station), ■ multihop (data is routed through the individual nodes toward the base station), and ■ clustering If the distance between the neighboring sensors is less than the average distance between the sensors and the user or the base station, transmission power can be saved if the sensors collaborate locally. Further, it's likely that sensors in local clusters share highly correlated data. Some of the nodes elect themselves as cluster heads and the remaining nodes join one of the clusters based on minimum transmission power criteria. The cluster head then aggregates and transmits the data from other cluster nodes. Such application-specific network protocols for wireless microsensor networks have been developed. They demonstrate that a clustering scheme is an order of magnitude more energy efficient than a simple direct transmission scheme. Power-aware sensor node model A power-aware sensor node model essentially describes the power consumption in different levels of node sleep state. Every component in the node can have different power modes. The StrongARM can be in active, idle, or sleep mode; the radio can be in transmit, receive, standby, or off mode. Each node sleep state corresponds to a particular combination of component power modes. In general, if there are N components labeled (1, 2, …, N) each with ki sleep states, the total number of node sleep states is ∏ki . Every component power mode has a latency overhead associated with transitioning to that mode. Therefore each node sleep mode is characterized by power consumption and latency overhead. However, from a practical point of view not all sleep states are useful. Table 1 enumerates the component power modes corresponding to five different useful sleep states for the sensor node. Each of these node sleep modes corresponds to an increasingly deeper sleep state and is therefore characterized by an increasing latency and decreasing power consumption. These sleep states are chosen based on actual working conditions of the sensor node; for example, it does not make sense to have memory active and everything else completely off. The design problem is to formulate a policy for transitioning between states based on observed events so as to maximize energy efficiency. The power-aware sensor model is similar to the system power model in the Advanced Configuration and Power Interface (ACPI) standard.7 An ACPI-compliant system has five global states. SystemStateS0 (corresponding to the working state), and SystemStateS1 to SystemStateS4 (corresponding to four different sleep-state levels). The sleep states are differentiated by power consumed, the overhead required in going to sleep and the wake-up time. In general, a deeper sleep state consumes less power and has a longer wake-up time. Another similar aspect is that in ACPI the power manager is an OS module. Event generation model An event occurs when a sensor node picks up a signal with power above a predetermined threshold. For analytical tractability, we assume that every node has a uniform radius of visibility, r. In real applications, the terrain might influence the visible radius. An event can be static (such as a localized change in temperature/pressure in an environment monitoring application) or can propagate (such as signals generated by a moving object in a tracking application). In general, events have a characterizable (possibly nonstationary) distribution in space and time. We will assume that the temporal

Moving average workload (MAW). The simplest filter is a time-invariant moving average filter, hn(k) = 1/N for all n and k. This filter predicts the workload in the next slot as the average of the workload in the previous N slots. The basic motivation is if the workload is truly an Nth-order Markov process, averaging will result in workload noise being removed by low-pass filtering. However, this scheme might be too simplistic and may not work with time-varying workload statistics. Also, averaging results in high-frequency workload changes are removed and as a result instantaneous performance hits are high. Exponential weighted averaging (EWA). This filter is based on the idea that the effect of a workload k slots before the current slot lessens as k increases. That is, this filter gives maximum weight to the previous slot, lesser weight to the one before, and so on. The filter coefficients are hn(k) = a−k , for all n, with a chosen so ∑hn(k) = 1 and is positive. The idea of exponential weighted averaging has been used in the prediction of idle times for DPM using shutdown techniques in event-driven computation. There, too, the idea is to assign progressively decreasing importance to historical data. Least mean square (LMS). It makes more sense to have an adaptive filter whose coefficients are modified based on the prediction error. Two popular adaptive filtering algorithms are the LMS and the recursive-least-squares (RLS) algorithms.9 The LMS adaptive filter is based on a stochastic gradient algorithm. Let the prediction error be $we(n) = w(n) - wp(n)$, where $we(n)$ denotes the error, and $w(n)$ denotes the actual workload as opposed to predicted workload $wp(n)$ from the previous slot. The filter coefficients are updated according to the following rule $hn+1(k) = hn(k) + \mu we(n) w(n - k)$ (12) where μ is the step size. Use of adaptive filters has its advantages and disadvantages. On the one hand, since they are self-designing, we do not have to worry about individual traces. The filters can learn from the workload history. The obvious problems involve convergence and stability. Choosing

5. Explain in detail about aggregation as an optimization problem. [CO4-L1]

We are interested in engineering optimization problems in which the physics are modeled using partial-differential equations. Specifically, we focus on problems of the form minimize $f(x,u)$ with respect to $x \in R\ n$, $u \in R\ m$ such that $g(\xi , x,u) \leq 1\ \forall \xi \in \Omega$ governed by $R(x,u) = 0$ Opt where $u \in R\ m$ denotes a finite-dimensional approximation of the solution to a partial differential equation (PDE), and $x \in R\ n$ is a vector of design variables. We will refer to u as the state variable and will adopt a reduced-space formulation; that is, the state will be considered an implicit function of the design variables via the discretized PDE $R(x,u) = 0$, where $R : R\ n \times R\ m \rightarrow R\ m$ is the residual. A distinguishing feature of the optimization problem Opt is the constraint $g(\xi , x,u) \leq 1$ that is imposed at all points within the (compact) domain Ω. This type of pointwise constraint arises in ∗Corresponding author, Assistant Professor, School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, email: graeme.kennedy@ae.gatech.edu †Assistant Professor, Department of Mechanical Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY, email: hickej2@rpi.edu 1 many problems of practical interest including stress- or displacement-constrained structural optimization [1, 23, 24, 29, 18, 11, 24], multidisciplinary aircraft design problems [19, 20], and some types of airfoil design problems [6]. Similar bound-constrained problems also occur in the context of time-domain simulations such as chemical process modeling [3, 25], satellite design [12], and composite manufacturing simulation [15]. The challenge presented by the constraint $g(\xi , x,u)$ is its infinite dimensionality. As a result, conventional, finite-dimensional optimization methods cannot be applied directly to solve problem Opt. This issue can be resolved by replacing the pointwise constraint with an equivalent, single constraint on the maximum, that is $\max \xi \in \Omega\ g(\xi , x,u) = maxg \leq 1$. (1) Unfortunately, a price is paid to reduce the infinite-dimensional constraint to a single constraint: a loss of smoothness. In general, changes to either x or u will cause the position and value of maxg to change, producing a continuous but non-differentiable functional output. Consequently, solving the optimization problem Opt with maxg in place of the continuous constraint $g \leq 1 \ \forall\xi \in \Omega$, is challenging for gradient-based design optimization methods that require sufficiently smooth constraint functions. While derivative-free optimization methods could accommodate this nonsmoothness, their computational cost scales poorly with the number of design variables and are therefore not effective for large-scale design problems. To address the issue of differentiability, practitioners often utilize constraint aggregation methods, in which the constraint maxg is approximated by a smooth function within the design optimization problem. To be consistent with the original problem Opt, this approximation must take the form:

$c(g,\rho) = maxg+r(g,\rho)$ (2)

where ρ is an aggregation parameter

that controls the functional approximation,

and $r(g,\rho)$ is a residual term.

The approximation must then satisfy the property

$\lim \rho\rightarrow\infty \ c(g,\rho) = maxg$, (3)

In other words,

the residual, $r(g,\rho)$,

should vanish as

$\rho \rightarrow \infty$.

Once an aggregation method has been selected, the original optimization problem Opt becomes the following, fully-discrete formulation: minimize $f(x,u)$ with respect to $x \in R \ n$, $u \in R \ m$ such that

$c(g(\xi , x,u),\rho) \leq 1$

governed by

$R(x,u) = 0$ AOpt(ρ)

One of the objectives of the present work is to assess the impact of different aggregation methods on

$x *(\rho)$, the solution of

AOpt(ρ).

In particular, we are interested in the accuracy of aggregation methods with respect to the limiting value

$\lim\rho\rightarrow\infty \ x *(\rho)$

and how well

$c(\rho, x,u)$

represents maxg. One way to assess the accuracy of an aggregation method is through its rate of convergence. For example, a method that is

first-order in $1/\rho$ satisfies

$c(g,\rho) = maxg+O \ \ 1 \ \rho$, (4) 2 where $O(\cdot)$

denotes the usual “big-oh” order notation. This first-order convergence rate is achieved by most practical aggregation methods when g is smooth. However, the coefficient of the dominant error term varies significantly between the different aggregation methods considered in this paper, so the rate of convergence is not an adequate measure of accuracy on its own. Bound-constrained problems of the form (Opt) have been solved by other authors using a variety of approaches. One common technique is to reformulate the non-differentiable maximum function by the addition of a slack variable and a series of bound constraints [27, 26, 2]. A key benefit of the bound formulation is that it eliminates the non-differentiable maximum constraint. Other authors have approached the optimization problem (Opt) by using smooth approximations of the bound constraint using constraint aggregation [18, 1, 23, 24, 29, 18, 11, 24]. The advantage of aggregation over the bound formulation is that no new additional constraints are required. In this paper, we present a new class of aggregation methods that are more accurate than classical methods. Solutions to problem AOpt(ρ) obtained with the proposed aggregation methods yield improved estimates of the solution to Opt. In this paper we will assess both the accuracy of aggregation methods, which estimate maxg, and the effect of the aggregation method on the solution of the optimization problem AOpt(ρ), $x*$ (ρ). In addition, we will use a first-order post-optimality sensitivity method to obtain an estimate of the design as $\rho \to \infty$. The remainder of this paper is structured as follows. In section 2, we introduce and analyze the convergence behavior of classical aggregation methods: the p-norm functional and the Kreisselmeier–Steinhauser (KS) functional. In section 3 we introduce a new class of method for constraint aggregation which we call induced aggregates. We demonstrate that some members of this class of aggregation functions yield more accurate estimates of maxg. In addition, we demonstrate a close relationship between the KS functional and a particular induced aggregation method. In practice, aggregation methods are applied to finite-dimensional problems using numerical quadrature schemes. In section 4, we examine the consequence of numerical quadrature on aggregation techniques. In section 5, we present a post-optimality analysis of optimization problems of the form AOpt(ρ). These enable a first-order prediction of the impact of the aggregation parameter ρ on the optimization solution. Finally, in section 6 we present the results of a series of studies which demonstrate the accuracy and effectiveness of the aggregation methods that we have proposed. We demonstrate the methods for a stress-constrained variable-thickness sheet problem and the design of a large aircraft wing. 1.1 Definitions, Assumptions, and Notation For simplicity, we assume that max$\xi \in \Omega$ g(ξ , x,u) exists for all (x,u) and is equal to the supremum of {g(ξ , x,u)|$\xi \in \Omega$}. Henceforth, we will drop the dependence of the max functional on Ω and omit the arguments to g to simplify the notation. An aggregation functional is conservative if the residual remains strictly positive, i.e. r(g,ρ) > 0, for all $\rho > \rho *$. This property ensures that the functional will yield an upper bound on maxg for sufficiently large ρ. Although most common aggregation functions are conservative in a discrete sense, they are not conservative within the context of the optimization problem Opt. For example, consider using the popular KS function to approximate the constraint g(ξ , x,u) ≤ 1 in Opt. This aggregation method relies on a finite set of trial locations, ξi ∈ Ω, i = 1,...,nt . One can show that the KS function will guarantee that g(ξi , x,u) ≤ 1, but it provides no assurances regarding feasibility at the remaining points Ω\{xi} nt i=1 . This deficiency is especially important in the context 3 of higher-order methods where significant solution variation may occur between trial locations. 2 Classical aggregation methods In this section, we present two classical aggregation functionals for the estimation of maxg: the KS and p-norm functionals. 2.1 The KS functional The KS functional is defined as cKS(g,ρ) = 1 ρ ln 1 α Z Ω e ρg dΩ = m+ 1 ρ ln 1 α Z Ω e ρ(g−m) dΩ , (5) where α is a normalization and m can be an arbitrary constant. We use the second, mathematically equivalent formulation in order to avoid issues with finite-precision arithmetic. Conventional normalizations for the KS function (defined below) and KS functional utilize a fixed value of α. Typical values are α = 1 and α = |Ω|, where |Ω| is the domain length, area, or volume. In the following, we consider arbitrary α and note that our selection will not interfere with the asymptotic convergence of the KS functional as long as lim ρ→∞ lnα ρ = 0. (6)

This requirement is derived from the limiting behavior of the KS functional1 : lim ρ→∞

cKS(g,ρ) = lim ρ→∞ 1
ρ lnZ Ω e
ρg dΩ − lnα
ρ = lim ρ→∞ 1
ρ lnZ Ω e
ρg dΩ = lim ρ→∞
ln||e g ||ρ = ln[maxe g] = maxg

Based on this observation, it is tempting to try to use α to achieve a conservative aggregate. To examine this strategy more closely, recall that the KS functional is conservative if it satisfies the condition:

cKS(g,ρ) ≥ maxg ∀ρ > ρ ∗ . (7)

Rearranging the expression for the KS functional we obtain the following requirement for conservativeness:

α ≤ Z Ω
e g emaxg
ρ dΩ = Z Ω e ρ(g−maxg) dΩ. (8)

1Recall the properties of the logarithm
ln(a/b) = ln(a)−ln(b) and μ ln(a) = ln(a μ).

4 Taking the natural logarithm of both sides of (8),
dividing by ρ, and taking the limit as ρ → ∞, we obtain (making use of ρ > 0 and the strict monotonicity of the natural logarithm): lim ρ→∞ lnα ρ ≤ lim ρ→∞ 1 ρ lnZ Ω e

ρ(g−maxg) dΩ = max(g−maxg) = 0, which satisfies requirement (6). However, selecting an α that satisfies this inequality is difficult. For instance, choosing α as a fixed fraction of the upper limit in (8), i.e. α = β Z Ω e ρ(g−maxg) dΩ, for β ≤ 1, would satisfy the inequality (8). However, the KS functional would then take the form: maxg− lnβ ρ , which would violate the differentiability requirement and defeat the purpose of using an aggregation technique. Nevertheless, the KS functional is conservative for certain functions. For instance, if the function attains the value maxg over a finite area of the domain then the functional will be conservative for any ρ as long as: α ≤ |{ξ ∈ Ω | g = maxg}|. (9) That is, the KS functional will be conservative if the scaling factor α is less than the size of the domain where g = maxg. However, if the region that the constraint achieves its maximum has measure zero in Ω — such as a point or a plane in three-dimensions — then it will not be possible to select α such that the KS functional is conservative. Examining the popular normalization, α = |Ω|, we note that this selection will be conservative only if g = maxg everywhere, that is, when g is a constant function. In all other cases, the KS functional with α = |Ω| is not conservative. For constant α < |Ω|, there is usually a small value of ρ such that the inequality (8) is satisfied, however this does not fully satisfy the conservative requirement, since the inequality (8) must be satisfied for all ρ > ρ ∗ . 2.1.1 Two illustrative examples To illustrate the behavior of the KS functional, consider the function g1(ξ) = ξ , ξ ∈ [0,1], such that maxg1 = 1. In this case, the KS functional takes the value: cKS(g1,ρ) = 1 ρ lnZ 1 0 e ρξ dξ = 1 ρ ln(e ρ −1)− lnρ ρ ≤ 1 where equality holds only as ρ → ∞. This example illustrates the non-conservative property for both α = 1 and α = |Ω|, since |Ω| = 1. As a second example, consider the KS functional of the piecewise function g2(ξ) = ξ 0 ≤ ξ ≤ 1, 1 1 < ξ ≤ 2, 5 for ξ ∈ [0,2]. The KS functional for the above function is: cKS(g2,ρ) = 1 ρ ln 1 αρ (e ρ −1) +e ρ = 1− lnα ρ + 1 ρ ln 1+ 1 ρ (1−e −ρ) , which is conservative for α ≤ 1. 2.1.2 The discrete KS function The discrete KS function is obtained by replacing the integral in the KS functional (5), with a summation over a finite set of constraints given by gi = g(ξi , x,u), where ξi , i = 1,...,nt , are the trial-point locations. The discrete KS function can then be written as follows: cDKS(ρ) = 1 ρ ln" 1 α nt ∑ i=1 e ρgi # = max i gi + 1 ρ ln" 1 α nt ∑ i=1 e ρ(gi−maxi gi) # . (10) If α is fixed, the discrete KS function grows without bound, even for constant ρ, as the number of trial locations increases. If instead we make a selection such as α = nt , then the KS function does remain bounded. Note, however, that the selection α = nt makes the argument to ln an approximate integral. 2.2 The p-norm functional In this paper, we write the p-norm functional as follows: cP(ρ) = 1 α Z Ω |g| ρ dΩ 1 ρ = m 1 α Z Ω   g m   ρ dΩ 1 ρ , (11) where α > 0 is a normalization factor and m > 0 is an otherwise arbitrary value. We use the second, mathematically equivalent form for computation, because it is less susceptible to numerical overflow. The requirement on the normalization is lim ρ→∞ 1 α 1 ρ = 1. This requirement is based on the limiting behavior of the functional: lim ρ→∞ cP(ρ) = lim ρ→∞ 1 α Z Ω |g| ρ dΩ 1 ρ = lim ρ→∞ 1 α 1 ρ lim ρ→∞ Z Ω |g| ρ dΩ 1 ρ = lim ρ→∞ Z Ω |g| ρ dΩ 1 ρ =

lim ρ→∞ ||g||ρ = ||g||∞ 6 The p-norm functional is conservative when: α ≤ Z Ω g maxg ρ dΩ (12) Again, the selection α = |Ω| is conservative only if g = max|g| everywhere within the domain. In all other cases, the p-norm functional with α = |Ω| is not conservative. For large values of ρ, the conservative property of the p-norm functional depends strongly on the behavior of g. Again, if the function g attains g = maxg on a measure-zero subspace of Ω, then it will not be possible to achieve a conservative estimate. Note that the p-norm functional estimates max|g| rather than maxg. As a result, the p-norm cannot be used to estimate the upper bound if the constraint g is unbounded below. However, in many situations, the range of the function g is known in advance, such that it is either non-negative, or has finite lower-bound. For example, quantities such as the Mach number or von Mises stress are physically limited to the range R+, while the Tsai–Wu criterion has finite, non-positive lower bound. In contrast, buckling envelopes are often unbounded from below, e.g. increasing a tensile load will produce a more negative buckling envelope function. Whenever g has a known finite lower-bound, it is possible to remap g so that it takes on only non-negative values. 2.3 The discrete p-norm function The discrete p-norm can be obtained by replacing the integral in the p-norm functional (11), by a discrete sum over a finite set of constraints given by gi = g(ξi , x,u), where, as before, ξi are a set of trial points locations. The discrete p-norm function can then be written as follows: cDP(ρ) = " 1 α ∑ i |gi | ρ # 1 ρ = max i |gi | " 1 α ∑ i gi maxi |gi | ρ # 1 ρ . (13) Like the discrete KS function, for fixed ρ, the discrete p-norm grows without bound as the number of trial locations increases. 3 Induced aggregation In this section, we examine a class of aggregation methods that are based on the approximate evaluation of a dual, or induced, norm. Like the classical aggregates described above, the proposed aggregation methods are not conservative; however, for the same value of ρ, they provide more accurate estimates of the upper bound maxg. We will demonstrate this accuracy in Section 6. These functional estimates can be used in the optimization problem AOpt(ρ) to obtain a tighter bound on the target optimization problem Opt. We begin by defining the aggregate operator. The purpose of this operator is to map g and ρ to a function that acts like a probability density function for the maximum value of g. Definition 3.1 (Aggregate Operator). An operator P : L 2 (Ω)×R → L 2 (Ω)is an aggregate operator if it maps a function g : L 2 (Ω) → R and parameter ρ > 0 to a new function that satisfies the following properties: 7 1. P(g(ξ),ρ) > 0; 2. R Ω P(g(ξ),ρ)dΩ = 1; 3. limρ→∞ P(g(ξ),ρ) = 0 wherever g(ξ) < maxg. From property 3, we see that an aggregate operator produces a function that is similar to a nascent Dirac delta function. Unlike a nascent delta function, which diverges at a single point only, P(g(ξ),ρ) tends to infinity on {ξ ∈ Ω | g(ξ) = maxg}, and this set may correspond to a point, curve, or subdomain of Ω. Using the aggregate operator, we can introduce a corresponding induced aggregate functional. Definition 3.2 (Induced Aggregate Functional). The induced aggregate functional for an aggregate operator P is the functional that maps the function g ∈ L 2 (Ω) and parameter ρ > 0 to the value cI(g,P,ρ) = Z Ω g(ξ)P(g(ξ),ρ)dΩ.

We now state and prove the paper’s main theoretical result. Theorem 3.1.

Let g ∈ L 2 (Ω)

be a function and

P : L 2 (Ω)×R → L 2 (Ω)

be an aggregate operator. Then the corresponding induced aggregate functional converges to the maximum value of g on the domain

Ω: lim ρ→∞ cI(g,P,ρ) = lim ρ→∞ Z

Ω g(ξ)P(g(ξ),ρ)dΩ = maxg.

Proof. We prove this theorem by constructing upper and lower bounds on cI whose limits are maxg. For the upper bound, it is easy to see that, for all

ρ > 0,

Z Ω g(ξ)P(g(ξ),ρ)dΩ ≤ (maxg)

Z Ω P(g(ξ),ρ)dΩ = maxg,

where we have used property 2 of the aggregate operator. To construct the lower bound, we first define the

subdomain(s) Ωε = {ξ ∈ Ω | g(ξ) > maxg−ε},

where ε > 0. Using Ωε ,

we find

Z Ω g(ξ)P(g(ξ),ρ)dΩ = Z Ωε

g(ξ)P(g(ξ),ρ)dΩ+ Z Ω\Ωε

g(ξ)P(g(ξ),ρ)dΩ ≥ (maxg−ε)

Z Ωε P(g(ξ),ρ)dΩ+ Z Ω\Ωε

g(ξ)P(g(ξ),ρ)dΩ

Next, we can use properties 2 and 3 of the aggregate operator to conclude that

lim ρ→∞ Z Ω\Ωε

g(ξ)P(g(ξ),ρ)dΩ = 0,

and

lim ρ→∞ (maxg−ε)

Z Ωε P(g(ξ),ρ)dΩ = (maxg−ε).

8 Summarizing, we have shown that

(maxg−ε) ≤ lim ρ→∞ Z Ω g(ξ)

P(g(ξ),ρ)dΩ ≤ maxg,

and since we can choose ε > 0 arbitrarily small we have

maxg ≤ lim ρ→∞ Z Ω g(ξ)P(g(ξ),ρ)dΩ ≤ maxg. The result now follows by the squeeze theorem.

6. Explain about the MAC protocol in WSN. [CO4-L2]

MPROVEMENTS in hardware technology have resulted in low-cost sensor nodes which are composed of a single chip with embedded memory, processor, and transceiver. Low power capacities lead to limited coverage and communication range for sensor nodes compared to other mobile devices. Hence, for example in target tracking and border surveillance applications, sensor networks must include a large number of nodes, to cover the target area successfully. Unlike other wireless networks, it is generally hard (or impractical) to charge/replace the exhausted battery, which gives way to the primary objective of maximizing node/network lifetime, leaving the other performance metrics as secondary objectives. Since the communication of sensor nodes will be more energy-consuming than their computation, it is a primary concern that the communication is minimized while achieving the desired network operation. However, the medium access decision within a dense network composed of nodes with low duty-cycles is a hard problem that must be solved in an energy-efficient manner. Having these in mind, Section II emphasizes the peculiar features of sensor networks including reasons of potential energy wastes at medium access communication. Then, Section III gives brief definitions for the key MAC protocols proposed for sensor networks listing their advantages and disadvantages. Moreover, the protocols that propose the integration of MAC layer with other layers are Ilker Demirkol, Cem Ersoy, and Fatih Alagöz are with the Network Research Laboratory (NETLAB) of the Computer Engineering Department of Bogazici University, Bebek, Istanbul, TURKEY (e-mail: {ilker,ersoy,alagoz}@boun.edu.tr). also investigated in Section III. Finally, Section IV concludes the survey on MAC protocols with a comparison of investigated protocols and provides a future direction to researchers for open issues that have not been studied thoroughly. II. MAC LAYER RELATED SENSOR NETWORK PROPERTIES Maximizing the network lifetime is a common objective of sensor network research, since sensor nodes are assumed to be disposed when they are out of battery. Under these circumstances, the proposed MAC protocol must be energyefficient by reducing the potential energy wastes presented in Section II.A. Types of communication patterns that are observed in sensor network applications should be investigated since these patterns are used to extract the behavior of the sensor network traffic that has to be handled by a given MAC protocol. Categorization of the possible communication patterns are outlined in Section II.B. Afterwards, the properties that must be possessed by a MAC protocol to suit a sensor network environment are presented in Section II.C. A. Reasons of Energy Waste When a receiver node receives more than one packet at the same time, these packets are called “collided packets” even when they coincide partially. All packets that cause the collision have to be discarded and the re-transmissions of these packets are required which increase the energy consumption. Although some packets could be recovered by a capture effect, a number of requirements have to be achieved for its success. The second reason of energy waste is overhearing, meaning that a node receives packets that are destined to other nodes. The third energy waste occurs as a result of control packet overhead. Minimal number of control packets should be used to make a data transmission. One of the major sources of energy waste is idle listening, i.e., listening to an idle channel to receive possible traffic. The last reason for energy waste is overemitting, which is caused by the transmission of a message when the destination node is not ready. Given the facts above, a correctly-designed MAC protocol should prevent these energy wastes. B. Communication Patterns Kulkarni et al. defines three types of communication patterns in wireless sensor networks [1]: broadcast, convergecast, and local gossip. Broadcast type of communication pattern is generally used by a base station (sink) to transmit some information to all sensor nodes of the network. Broadcasted information may include queries MAC Protocols for Wireless Sensor Networks: a Survey Ilker Demirkol, Cem Ersoy, and Fatih Alagöz I of sensor query-processing architectures, program updates for sensor nodes, control packets for the whole system. The broadcast type communication pattern should not be confused with broadcast type packet. For the former, all nodes of the network are intended receivers whereas for the latter the intended receivers are the nodes within the communication range of the transmitting node. In some scenarios, the sensors that detect an intruder communicate with each other locally. This kind of communication pattern is called local gossip, where a sensor sends a message to its neighboring nodes within a range. The sensors that detect the intruder, then, need to send what they perceive to the information center. That communication pattern is called convergecast, where a group of sensors communicate to a specific sensor. The destination node could be a clusterhead, data fusion center, base station. In protocols that include clustering, clusterheads communicate with their members and thus the intended receivers may not be all neighbors of the clusterhead, but just a subset of the neighbors. To serve for such scenarios, we define a fourth type of communication pattern, multicast, where a sensor sends a message to a specific subset of sensors. C. Properties of a Well-defined MAC Protocol To design a good MAC protocol for the wireless sensor networks, the following attributes must be considered [2]. The first attribute is the energy efficiency. We have to define energy efficient protocols in order to prolong the network lifetime. Other important attributes are scalability and adaptability to changes. Changes in network size, node density and topology should be handled rapidly and effectively for a successful adaptation. Some of the reasons behind these network property changes are limited node lifetime, addition of new nodes to the network and varying interference which may alter the connectivity and hence the network topology. A good MAC protocol should gracefully accommodate such network changes. Other typical important attributes such as latency, throughput and bandwidth utilization may be secondary in sensor networks. Contrary to other wireless networks, fairness among sensor nodes is not usually a design goal, since all sensor nodes share a common task. III. PROPOSED MAC LAYER PROTOCOLS In this section, a wide range of MAC protocols defined for sensor networks are described briefly by stating the essential behavior of the protocols wherever possible. Moreover, the advantages and disadvantages of these protocols are presented. 1) Sensor-MAC (S-MAC) Locally managed synchronizations and periodic sleeplisten schedules based on these synchronizations form the basic idea behind the Sensor-MAC (S-MAC) protocol [2]. Neighboring nodes form virtual clusters to set up a common sleep schedule. If two neighboring nodes reside in two different virtual clusters, they wake up at listen periods of both clusters. A drawback of S-MAC algorithm is this possibility of following two different schedules, which results in more energy consumption via idle listening and overhearing. Schedule exchanges are accomplished by periodical SYNC packet broadcasts to immediate neighbors. The period for each node to send a SYNC packet is called the synchronization period. Figure 1 represents a sample sender-receiver communication. Collision avoidance is achieved by a carrier sense, which is represented as CS in the figure. Furthermore, RTS/CTS packet exchanges are used for unicast type data packets. An important feature of S-MAC is the concept of message-passing where long messages are divided into frames and sent in a burst. With this technique, one may achieve energy savings by minimizing communication overhead at the expense of unfairness in medium access. Periodic sleep may result in high latency especially for multi-hop routing algorithms, since all immediate nodes have their own sleep schedules. The latency caused by periodic sleeping is called sleep delay in [2]. Adaptive listening technique is proposed to improve the sleep delay, and thus the overall latency. In that technique, the node who overhears its neighbor’s transmissions wakes up for a short time at the end of the transmission. Hence, if the node is the next-hop node, its neighbor could pass data immediately. The end of the transmissions is known by the duration field of RTS/CTS packets. Figure 1. S-MAC Messaging Scenario [2] Advantages: The energy waste caused by idle listening is reduced by sleep schedules. In addition to its implementation simplicity, time synchronization overhead may be prevented with sleep schedule announcements. Disadvantages: Broadcast data packets do not use RTS/CTS which increases collision probability. Adaptive listening incurs overhearing or idle listening if the packet is not destined to the listening node. Sleep and listen periods are predefined and constant, which decreases the efficiency of the algorithm under variable traffic load. 2) WiseMAC Spatial TDMA and CSMA with Preamble Sampling protocol is proposed in [3] where all sensor nodes are defined to have two communication channels. Data channel is accessed with TDMA method, whereas the control channel is accessed with CSMA method. Enz et al. proposed WiseMAC [4] protocol which is similar to Hoiydi et al.’s work [3] but requires only a single-channel. WiseMAC protocol uses non-persistent CSMA (np-CSMA) with preamble sampling as in [3] to decrease idle listening. In the preamble sampling technique, a preamble precedes 3 each data packet for alerting the receiving node. All nodes in a network sample the medium with a common period, but their relative schedule offsets are independent. If a node finds the medium busy after it wakes up and samples the medium, it continues to listen until it receives a data packet or the medium becomes idle again. The size of the preamble is initially set to be equal to the sampling period. However, the receiver may not be ready at the end of the preamble, due to reasons like interference, which causes the possibility of overemitting type energy waste. Moreover, overemitting is increased with the length of the preamble and the data packet, since no handshake is done with the intended receiver. To reduce the power consumption incurred by the predetermined fixed-length preamble, WiseMAC offers a method to dynamically determine the length of the preamble. That method uses the knowledge of the sleep schedules of the transmitter node’s direct neighbors. The nodes learn and refresh their neighbor’s sleep schedule during every data exchange as part of the acknowledgement message. In that way, every node keeps a table of sleep schedules of its neighbors. Based on neighbors’ sleep schedule table, WiseMAC schedules transmissions so that the destination node’s sampling time corresponds to the middle of the sender’s preamble. To decrease the possibility of collisions caused by that specific start time of wake-up preamble, a random wake-up preamble is advised. Another parameter affecting the choice of the wake-up preamble length is the potential clock drift between the source and the destination. A lower bound for the preamble length is calculated as the minimum of destination’s sampling period, Tw, and the potential clock drift with the destination which is a multiple of the time since the last ACK packet arrival. Considering this lower bound, a preamble length, Tp, is chosen randomly. Figure 2 presents the WiseMAC concept. Figure 2. WiseMAC Concept [4] Advantages: The simulation results show that WiseMAC performs better than one of the S-MAC variants [4]. Besides, its dynamic preamble length adjustment results in better performance under variable traffic conditions. In addition, clock drifts are handled in the protocol definition which mitigates the external time synchronization requirement. Disadvantages: Main drawback of WiseMAC is that decentralized sleep-listen scheduling results in different sleep and wake-up times for each neighbor of a node. This is especially an important problem for broadcast type of communication, since broadcasted packet will be buffered for neighbors in sleep mode and delivered many times as each neighbor wakes up. However, this redundant transmission will result in higher latency and power consumption. In addition, the hidden terminal problem comes along with WiseMAC model as in the Spatial TDMA and CSMA with Preamble Sampling algorithm. That is because WiseMAC is also based on non-persistent CSMA. This problem will result in collisions when one node starts to transmit the preamble to a node that is already receiving another node’s transmission where the preamble sender is not within the range. 3) Traffic-Adaptive MAC Protocol (TRAMA) TRAMA [5] is a TDMA-based algorithm and proposed to increase the utilization of classical TDMA in an energyefficient manner. It is similar to Node Activation Multiple Access (NAMA) [6], where for each time slot a distributed election algorithm is used to select one transmitter within two-hop neighborhood. This kind of election eliminates the hidden terminal problem and hence, ensures all nodes in the one-hop neighborhood of the transmitter will receive data without any collision. However, NAMA is not energyefficient, and incurs overhearing. Time is divided into random-access and scheduled-access (transmission) periods. Random-access period is used to establish two-hop topology information where channel access is contention-based. A basic assumption is that, by the information passed by the application layer, MAC layer can calculate the transmission duration needed which is denoted as SCHEDULE_INTERVAL. Then at time t, the node calculates the number of slots for which it will have the highest priority among two-hop neighbors within the period [t,t+ SCHEDULE_INTERVAL]. The node announces the slots it will use as well as the intended receivers for these slots with a schedule packet. Additionally, the node announces the slots for which it has the highest priority but will not be used. The schedule packet indicates the intended receivers using a bitmap whose length is equal to the number of its neighbors. Bits correspond to one-hop neighbors ordered by their identities. Since the receivers of those messages have the exact list and identities of the onehop neighbors, they find out the intended receiver. When the vacant slots are announced, potential senders are evaluated for re-use of those slots. Priority of a node on a slot is calculated with a hash function of node’s and slot’s identities. Analytical models for the delay performances of TRAMA and NAMA protocols are also presented and supported by simulations [5]. Delays are found to be higher compared to contention-based protocols due to higher percentage of sleep times. Advantages: Higher percentage of sleep time and less collision probability is achieved compared to CSMA based protocols. Since intended receivers are indicated with a bitmap, less communication is performed for multicast and broadcast type of communication patterns compared other 4 protocols. Disadvantages: Transmission slots are set to be seven times longer than the random access period [5]. However, all nodes are defined to be either in receive or transmit states during the random access period for schedule exchanges. This means that without considering the transmissions and receptions, the duty cycle is at least 12.5 %, which is a considerably high value. For a time slot, every node calculates each of its two-hop neighbors’ priorities on that slot. In addition, this calculation is repeated for each time slot, since the parameters of the calculation change with time. 4) SIFT Sift [7] is a MAC protocol proposed for event-driven sensor network environments. The motivation behind Sift is that when an event is sensed, the first R of N potential reports is the most crucial part of messaging and has to be relayed with low latency. Jamieson et al. use a non-uniform probability distribution function of picking a slot within the slotted contention window. If no node starts to transmit in the first slot of the window, then each node increases its transmission probability exponentially for the next slot assuming that the number of competing nodes is small. In [7], Sift is compared with 802.11 MAC protocol and it is showed that Sift decreases latency considerably when there are many nodes trying to send a report. Since Sift is a method for contention slot assignment algorithm, it is proposed to co-exist with other MAC protocols like SMAC. Based on the same idea, CSMA/p* is proposed in

[8] where p* is a non-uniform probability distribution that optimally minimizes latency. However, Tay et al. state that Sift has a distribution approximate to CSMA/p*. Advantages: Very low latency is achieved with many traffic sources. Energy consumption is traded off for latency as indicated below. However, when the latency is an important parameter of the system, slightly increased energy consumption must be accepted. It could be tuned to incur less energy consumption. The high energy consumption is a result of the arguments indicated below. Disadvantages: One of the main drawbacks is increased idle listening caused by listening to all slots before sending. The second drawback is increased overhearing. When there is an ongoing transmission, nodes must listen till the end in order to contend for the next transmission which causes overhearing. Besides, system-wide time synchronization is needed for slotted contention windows. That is why, the implementation complexity of Sift would be increased for the protocols not utilizing time synchronization. 5) DMAC Convergecast is the mostly observed communication pattern within sensor networks. These unidirectional paths from possible sources to the sink could be represented as data gathering trees. The principal aim of DMAC [9] is to achieve very low latency, but still to be energy efficient. DMAC could be summarized as an improved Slotted Aloha algorithm where slots are assigned to the sets of nodes based on a data gathering tree as shown in Figure 3. Hence, during the receive period of a node, all of its child nodes has transmit periods and contend for the medium. Low latency is achieved by assigning subsequent slots to the nodes that are successive in the data transmission path. sleep sleep Rx Tx sleep sleep sleep Rx Tx Rx Tx Rx Tx Rx Tx Rx Tx Rx Tx Rx Tx Rx Tx Rx Tx Fig. 3. A data gathering tree and its DMAC implementation [9] Advantages: DMAC achieves very good latency compared to other sleep/listen period assignment methods. The latency of the network is crucial for certain scenarios, in which DMAC could be a strong candidate. Disadvantages: Collision avoidance methods are not utilized, hence when a number of nodes that has the same schedule (same level in the tree) try to send to the same node, collisions will occur. This is a possible scenario in event-triggered sensor networks. Besides, the data transmission paths may not be known in advance, which precludes the formation of the data gathering tree. 6) Timeout-MAC (T-MAC) / Dynamic Sensor-MAC (DSMAC) Static sleep-listen periods of S-MAC result in high latency and lower throughput as indicated earlier. TimeoutMAC (T-MAC) [10] is proposed to enhance the poor results of S-MAC protocol under variable traffic load. In T-MAC, listen period ends when no activation event has occurred for a time threshold TA. The decision for TA is presented along with some solutions to the early sleeping problem defined in [10]. Variable load in sensor networks are expected, since the nodes that are closer to the sink must relay more traffic. Although T-MAC gives better results under these variable loads, the synchronization of the listen periods within virtual clusters is broken. This is one of the reasons for the early sleeping problem. Dynamic Sensor-MAC (DSMAC) [11] adds dynamic duty cycle feature to S-MAC. The aim is to

decrease the latency for delay-sensitive applications. Within the SYNC period, all nodes share their one-hop latency values (time between the reception of a packet into the queue and its transmission). All nodes start with the same duty cycle. Figure 4 conceptually depicts DSMAC duty cycle doubling. When a receiver node notices that average one-hop latency value is high, it decides to shorten its sleep time and announces it within SYNC period. Accordingly, after a sender node receives this sleep period decrement signal, it checks its queue for packets destined to that receiver node. If there is one, it decides to double its duty cycle when its battery level is above a specified threshold. 5 Fig 4. DSMAC duty cycle doubling [11] The duty cycle is doubled so that the schedules of the neighbors will not be affected. The latency observed with DSMAC is better than the one observed with S-MAC. Moreover, it is also shown to have better average power consumption per packet. 7) Integration of MAC with Other Layers Limited research has been carried out to integrate different network layers to one layer or to benefit from the cross-layer interactions between routing and MAC layers for sensor networks. One such research is done by Safwat et al. who proposed two routing algorithms that favor the information about successful/unsuccessful CTS or ACK reception [12]. Cui et al. have research in that area with the objectives of MAC/Physical layer integration and Routing/MAC/Physical layer integration [13]. They propose a variable length TDMA scheme where the slot length is assigned according to some criteria for the optimum energy consumption in the network. Among these criteria, the most crucial ones are information about the traffic generated by each node and distances between each node pair. Based on these values, they formulate a Linear Programming (LP) problem where the decision variables are normalized time slot lengths between nodes. They solve this LP problem using an LP solver which returns the optimum number of time slots for each node pairs as well as the related routing decisions for the system. The proposed solution could be quite beneficial for scenarios where the required data could be prepared. However, it is generally hard to have the node distance information and the traffic generated by the nodes. Besides, LP solver could only be run on a powerful node. However, the dynamic behaviors of sensor networks will require online decisions which are very costly to calculate and hard to adapt to an existing system. Multihop Infrastructure Network Architecture (MINA) is another work for integrating MAC and routing protocols [14]. Ding et al. propose a layered multi-hop network architecture where the network nodes with the same hopcount to the base station are grouped into the same layer. Channel access is a TDMA-based MAC protocol combined with CDMA or FDMA. The super-frame is composed of a control packet, a beacon frame and a data transmission frame. Beacon and data frames are time slotted. In the clustered network architecture, all members of a cluster submit their transmission requests in beacon slots. Accordingly, the cluster-head announces the schedule of the data frame. The routing protocol is a simple multi-hop protocol where each node has a forwarder node at one nearer layer to the base station.

The forwarding node is chosen from candidates based on the residual energies. Ding et al. then formulate the channel allocation problem as an NP-complete problem and propose a sub-optimal solution. Moreover, the transmission range of sensor nodes is a decision variable, since it affects the layering of the network (hop-counts change). Simulations are run to find a good range of values for a specific scenario. The proposed system in [14] is a well-defined MAC/Routing system. However, the tuning of the range parameter is an important task which should be determined at the system initialization. In addition, all node-to-sink paths are defined at the startup and are defined to be static, since channel frequency assignments of nodes are done at the startup accordingly. This makes the system intolerant to failures. Geographic Random Forwarding (GeRaF) is actually proposed as a routing protocol, but the underlying MAC algorithm is also defined in the work which is based on CSMA/CA [15]. That gives us not integrated but a complete solution for a sensor network's communication layers. The difficulty of the system proposed is its need for additional radio, which is used for busy tone announcement. Rugin et al. [16] and Zorzi et al. [15] improved GeRaF reducing it to a one-channel system. However, sensor nodes' and their neighbors' location information are needed for those protocols. Besides, the forwarding node is chosen among nodes that are awake at the time of the transmission request. That may result in more power consuming routing, and an increase in latency.

7. Write in detail about the IEEE 802.15.4 [CO4-L1]

IEEE standard 802.15.4 intends to offer the fundamental lower network layers of a type of wireless personal area network (WPAN) which focuses on low-cost, low-speed ubiquitous communication between devices. It can be contrasted with other approaches, such as Wi-Fi, which offer more bandwidth and require more power. The emphasis is on very low cost communication of nearby devices with little to no underlying infrastructure, intending to exploit this to lower power consumption even more.

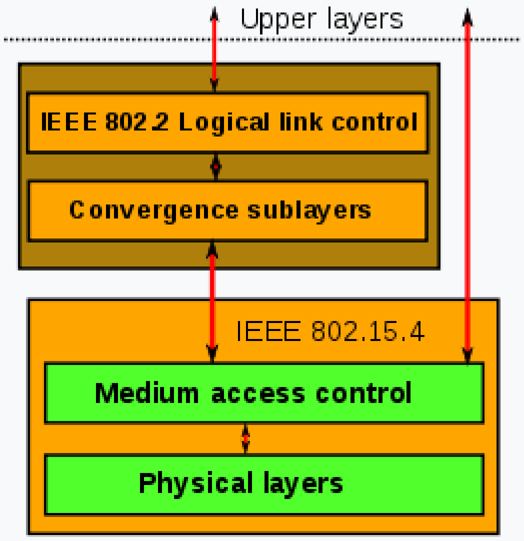
The basic framework conceives a 10-meter communications range with a transfer rate of 250 kbit/s. Tradeoffs are possible to favor more radically embedded devices with even lower power requirements, through the definition of not one, but several physical layers. Lower transfer rates of 20 and 40 kbit/s were initially defined, with the 100 kbit/s rate being added in the current revision.

Even lower rates can be considered with the resulting effect on power consumption. As already mentioned, the main identifying feature of IEEE 802.15.4 among WPANs is the importance of achieving extremely low manufacturing and operation costs and technological simplicity, without sacrificing flexibility or generality.

Important features include real-time suitability by reservation of guaranteed time slots, collision avoidance through CSMA/CA and integrated support for secure communications. Devices also include power management functions such as link quality and energy detection.

IEEE 802.15.4-conformant devices may use one of three possible frequency bands for operation (868/915/2450 MHz).

Protocol architecture



IEEE 802.15.4 protocol stack

Devices are conceived to interact with each other over a conceptually simple wireless network. The definition of the network layers is based on the OSI model; although only the lower layers are defined in the standard, interaction with upper layers is intended, possibly using an IEEE 802.2 logical link control sublayer accessing the MAC through a convergence sublayer. Implementations may rely on external devices or be purely embedded, self-functioning devices.

The physical layer

The physical layer is the initial layer in the OSI reference model used worldwide. The *physical layer* (PHY) ultimately provides the data transmission service, as well as the interface to the *physical layer management entity*, which offers access to every layer management function and maintains a database of information on related personal area networks. Thus, the PHY manages the physical RF transceiver and performs channel selection and energy and signal management functions. It operates on one of three possible unlicensed frequency bands:

- 868.0–868.6 MHz: Europe, allows one communication channel (2003, 2006, 2011[4])
- 902–928 MHz: North America, up to ten channels (2003), extended to thirty (2006)
- 2400–2483.5 MHz: worldwide use, up to sixteen channels (2003, 2006)

The original 2003 version of the standard specifies two physical layers based on *direct sequence spread spectrum* (DSSS) techniques: one working in the 868/915 MHz bands with transfer rates of 20 and 40 kbit/s, and one in the 2450 MHz band with a rate of 250 kbit/s.

The 2006 revision improves the maximum data rates of the 868/915 MHz bands, bringing them up to support 100 and 250 kbit/s as well. Moreover, it goes on to define four physical layers depending on the modulation method used. Three of them preserve the DSSS approach: in the 868/915 MHz bands, using either binary or offset quadrature phase shift keying (the second of which is optional); in the 2450 MHz band, using the latter. An alternative, optional 868/915 MHz layer is defined using a combination of binary keying and amplitude shift keying (thus based on parallel, not sequential spread spectrum, PSSS). Dynamic switching between supported 868/915 MHz PHYs is possible.

Beyond these three bands, the IEEE 802.15.4c study group considered the newly opened 314–316 MHz, 430–434 MHz, and 779–787 MHz bands in China, while the IEEE 802.15 Task Group 4d defined an amendment to 802.15.4-2006 to support the new 950–956 MHz band in Japan. First standard amendments by these groups were released in April 2009.

In August 2007, IEEE 802.15.4a was released expanding the four PHYs available in the earlier 2006 version to six, including one PHY using Direct Sequence ultra-wideband (UWB) and another using chirp spread spectrum (CSS). The UWB PHY is allocated frequencies in three ranges: below 1 GHz, between 3 and 5 GHz, and between 6 and 10 GHz. The CSS PHY is allocated spectrum in the 2450 MHz ISM band.[5]

In April, 2009 IEEE 802.15.4c and IEEE 802.15.4d were released expanding the available PHYs with several additional PHYs: one for 780 MHz band using O-QPSK or MPSK,[6]another for 950 MHz using GFSK or BPSK.[7]

IEEE 802.15.4e was chartered to define a MAC amendment to the existing standard 802.15.4-2006 which adopts channel hopping strategy to improve support for the industrial markets, increases robustness against external interference and persistent multi-path fading. On February 6, 2012 the IEEE Standards Association Board approved the IEEE 802.15.4e which concluded all Task Group 4e efforts.

The MAC layer

The medium access control (MAC) enables the transmission of MAC frames through the use of the physical channel. Besides the data service, it offers a management interface and itself manages access to the physical channel and network beaconing. It also controls frame validation, guarantees time slots and handles node associations. Finally, it offers hook points for secure services.

Note that the IEEE 802.15 standard does *not* use 802.1D or 802.1Q, i.e., it does not exchange standard Ethernet frames. The physical frame-format is specified in IEEE802.15.4-2011 in section 5.2. It is tailored to the fact that most IEEE 802.15.4 PHYs only support frames of up to 127 bytes (adaptation layer protocols such as 6LoWPAN provide fragmentation schemes to support larger network layer packets).

Higher layers

Other higher-level layers and interoperability sublayers are not defined in the standard. Specifications, such as 6LoWPAN and ZigBee, build on this standard. RIOT, TinyOS, Unison RTOS, DSPnano RTOS, nanoQplus and Contiki operating systems also use a few items of IEEE 802.15.4 hardware and software.

Network model

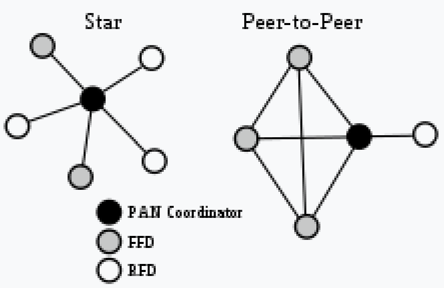
Node types

The standard defines two types of network node.

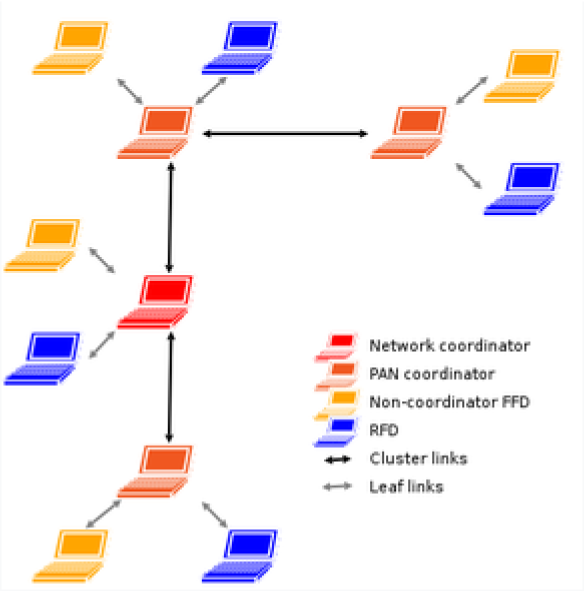
The first one is the **full-function device** (FFD). It can serve as the coordinator of a personal area network just as it may function as a common node. It implements a general model of communication which allows it to talk to any other device: it may also relay messages, in which case it is dubbed a coordinator (PAN coordinator when it is in charge of the whole network).

On the other hand, there are **reduced-function devices** (RFD). These are meant to be extremely simple devices with very modest resource and communication requirements; due to this, they can only communicate with FFDs and can never act as coordinators.

Topologies



IEEE 802.15.4 star and peer-to-peer



IEEE 802.15.4 cluster tree

Networks can be built as either peer-to-peer or star networks. However, every network needs at least one FFD to work as the coordinator of the network. Networks are thus formed by groups of devices separated by suitable distances. Each device has a unique 64-bit identifier, and if some conditions are met short 16-bit identifiers can be used within a restricted environment. Namely, within each PAN domain, communications will probably use short identifiers.

Peer-to-peer (or point-to-point) networks can form arbitrary patterns of connections, and their extension is only limited by the distance between each pair of nodes. They are meant to serve as the basis for ad hoc networks capable of performing self-management and organization. Since the standard does not define a network layer, routing is not directly supported, but such an additional layer can add support for multihop communications. Further topological restrictions may be added; the standard mentions the cluster tree as a structure which exploits the fact that an RFD may only be associated with one FFD at a time to form a network where RFDs are exclusively leaves of a tree, and most of the nodes are FFDs. The structure can be extended as a generic mesh network whose nodes are cluster tree networks with a local coordinator for each cluster, in addition to the global coordinator.

A more structured **star** pattern is also supported, where the coordinator of the network will necessarily be the central node. Such a network can originate when an FFD decides to create its own PAN and declare itself its coordinator, after choosing a unique PAN identifier. After that, other devices can join the network, which is fully independent from all other star networks.

Data transport architecture

Frames are the basic unit of data transport, of which there are four fundamental types (data, acknowledgment, beacon and MAC command frames), which provide a reasonable tradeoff between simplicity and robustness. Additionally, a superframe structure, defined by the coordinator, may be used, in which case two beacons act as its limits and provide synchronization to other devices as well as configuration information. A superframe consists of sixteen equal-length slots, which can be further divided into an active part and an inactive part, during which the coordinator may enter power saving mode, not needing to control its network.

Within superframes contention occurs between their limits, and is resolved by CSMA/CA. Every transmission must end before the arrival of the second beacon. As mentioned before, applications with well-defined bandwidth needs can use up to seven domains of one or more contentionless guaranteed time slots, trailing at the end of the superframe. The first part of the superframe must be sufficient to give service to the network structure and its devices. Superframes are typically utilized within the context of low-latency devices, whose associations must be kept even if inactive for long periods of time.

Data transfers to the coordinator require a beacon synchronization phase, if applicable, followed by CSMA/CA transmission (by means of slots if superframes are in use); acknowledgment is optional. Data transfers from the coordinator usually follow device requests: if beacons are in use, these are used to signal requests; the coordinator acknowledges the request and then sends the data in packets which are acknowledged by the device. The same is done when superframes are not in use, only in this case there are no beacons to keep track of pending messages.

Point-to-point networks may either use unslotted CSMA/CA or synchronization mechanisms; in this case, communication between any two devices is possible, whereas in "structured" modes one of the devices must be the network coordinator.

In general, all implemented procedures follow a typical request-confirm/indication-response classification.

Reliability and security

The physical medium is accessed through a CSMA/CA protocol. Networks which are not using beaconing mechanisms utilize an unslotted variation which is based on the listening of the medium, leveraged by a random exponential backoff algorithm; acknowledgments do not adhere to this discipline. Common data transmission utilizes unallocated slots when beaconing is in use; again, confirmations do not follow the same process.

Confirmation messages may be optional under certain circumstances, in which case a success assumption is made. Whatever the case, if a device is unable to process a frame at a given time, it simply does not confirm its reception: timeout-based retransmission can be performed a number of times, following after that a decision of whether to abort or keep trying.

Because the predicted environment of these devices demands maximization of battery life, the protocols tend to favor the methods which lead to it, implementing periodic checks for pending messages, the frequency of which depends on application needs.

Regarding secure communications, the MAC sublayer offers facilities which can be harnessed by upper layers to achieve the desired level of security. Higher-layer processes may specify keys to perform symmetric cryptography to protect the payload and restrict it to a group of devices or just a point-to-point link; these groups of devices can be specified in access control lists. Furthermore, MAC computes *freshness checks* between successive receptions to ensure that presumably old frames, or data which is no longer considered valid, does not transcend to higher layers.

In addition to this secure mode, there is another, insecure MAC mode, which allows access control lists[2] merely as a means to decide on the acceptance of frames according to their (presumed) source.

Unit –V

WSN ROUTING, LOCALIZATION & QOS

Part – A

1. How the sensor application data models can be arise? [CO5-L1]

Wireless sensor networks are an emerging area of research interest with a number of compelling potential applications. By architecting sensor networks as virtual databases, we can provide a well-understood nonprocedural programming interface suitable to data management, allowing the community to realize sensornet applications rapidly

2. Define: OLSR [CO5-L2]

The Optimized Link State Routing Protocol (OLSR)[1] is an IP routing protocol optimized for mobile ad hoc networks, which can also be used on other wireless ad hoc networks. OLSR is a proactive link-state routing protocol, which uses *hello* and *topology control* (TC) messages to discover and then disseminate link state information throughout the mobile ad hoc network. Individual nodes use this topology information to compute next hop destinations for all nodes in the network using shortest hop forwarding paths.

3. What are the advantages of the OLSR protocol? [CO5-L1]

Being a proactive protocol, routes to all destinations within the network are known and maintained before use. Having the routes available within the standard routing table can be useful for some systems and network applications as there is no route discovery delay associated with finding a new route.

The routing overhead generated, while generally greater than that of a reactive protocol, does not increase with the number of routes being created.

4. Write short notes on localization[CO5-L2]

In this Section we review the localization formula due to Berline-Vergne [BV] and Atiyah-Bott [AB]. It is then used to derive the exact stationary phase formula due to Duistermaat and Heckman [DH]. The presentation is illustrated at the elementary example of sphere S2

5. What is called beacons? [CO5-L1]

Beacon is an intentionally conspicuous device designed to attract attention to a specific location.

Beacons can also be combined with semaphoric or other indicators to provide important information, such as the status of an airport, by the colour and rotational pattern of its airport beacon, or of pending weather as indicated on a weather beacon mounted at the top of a tall building or similar site. When used in such fashion, beacons can be considered a form of optical telegraphy.

6. How the indoor localization works? [CO5-L1]

An indoor positioning system (IPS) is a system to locate objects or people inside a building using radio waves, magnetic fields, acoustic signals, or other sensory information collected by mobile devices. There are several commercial systems on the market, but there is no standard for an IPS system.

IPSes use different technologies, including distance measurement to nearby anchor nodes (nodes with known positions, e.g., WiFi access points), magnetic positioning, dead reckoning. They either actively locate mobile devices and tags or provide ambient location or environmental context for devices to get sensed.

7. Define: Lateration [CO5-L2]

The focus of this research is to develop a technique for indoor localization for tracking activity using Wireless Local Area Network (WLAN). The research is conducted by studying the characteristic of the WiFi signal propagation inside a multi-storey building during static condition and compare with walking conditions to compare the accuracy of the tracking technique.

8. Define: Angulation [CO5-L2]

To transition from bivalent to monovalent step, must exist a critical step which allows the transition by making a decision Critical step cannot be local (cannot tell apart between slow and failed process) nor can it be across multiple processes (it would not be well-defined) Hence, cannot transit from bivalent to univalent state.

9. How to determine the distance in triangulation? [CO5-L1]

This article is about measurement by the use of triangles. For other uses, see Triangulation (disambiguation).

In trigonometry and geometry, triangulation is the process of determining the location of a point by forming triangles to it from known points.

Specifically in surveying, triangulation per se involves only angle measurements, rather than measuring distances to the point directly as in trilateration; the use of both angles and distance measurements is referred to as triangulateration.

10. Define: Physical time [CO5-L2]

By showing reduction from consensus to problem X, then X is also not solvable under same model (single crash failure) E.g., leader election, terminating reliable broadcast, atomic broadcast, computing a network-wide global function using BC-CC flows, transaction commit.

11. Define: Synchronous agreement [CO5-L2]

Validity: If the sender of a broadcast message m is non-faulty, then all correct processes eventually deliver m. Agreement: If a correct process delivers a message m, then all correct processes deliver m. Integrity: Each correct process delivers at most one message. Further, if it delivers a message different from the null message, then the sender must have broadcast m. Termination: Every correct process eventually delivers some message.

12. What is the two domain classification in Qos? [CO5-L2]

Match Command	Match Criteria
match cos	Matches based on traffic with a specific Class of Service (CoS) value
match dscp	Matches based on traffic with a specific Differentiated Services Code Point (DSCP) value
match precedence	Matches based on traffic with a specific IP precedence value

13. What are the challenges in Qos? [CO5-L1]

1. Too weak a proxy for user outcomes.
2. Obsessed with real-time flows.
3. Misallocates resources.
4. Opens a new denial of service attack route.
5. “Quality inversion” removes the economic incentive to pay.
6. Too limited in the scope of the trades available.

14. What is energy efficient design? [CO5-L2]

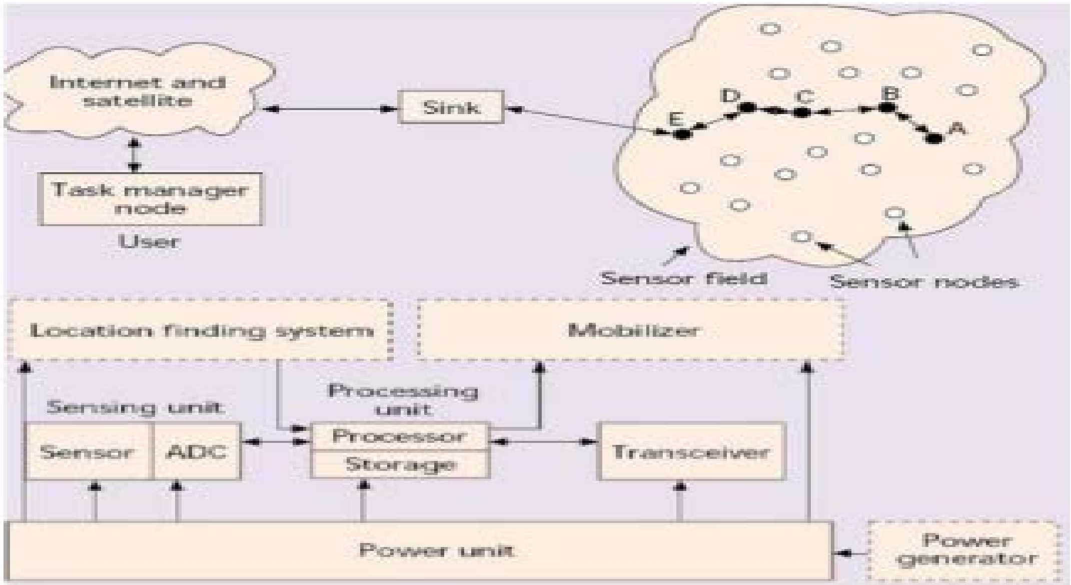
Energy Efficient Design (EED) is a methodology that assists organizations to design, construct and manage projects to achieve minimum energy consumption. This approach can help Public Sector bodies to future-proof projects now being considered, as part of their 2020 Energy Reduction commitments.

15. Write short notes on design factor of transport protocol? [CO5-L1]

Providing efficient transport services over multi-hop ad hoc networks is a fundamental building block for this wireless technology. The typical approach is modifying TCP to fix one (or a few of) its inefficiency while preserving compatibility with the original protocol. However, a complete solution should include a significant number of modifications, such that the original TCP design is deeply modified. In this paper we explore a different approach. We include the desired modifications to TCP in the design of a new transport protocol (TPA).

Part – B

1. Briefly explain the issues in WSN routing [CO5-L1]



Due to advance information technology, Wireless sensor networks (WSN's) are rapidly developing area in both research and application. The wireless sensor networks are based on the co-operation of a number of tiny sensors and which are depending upon the four parts: sensor (motes), processor, transceiver, and battery. The Sensor get information from surrounding area and processor change the analog information into digital information. Wireless sensors have the ability to perform simple calculations and communicate in a small area. Wireless sensor networks have critical applications in the scientific, medical, commercial, and military domains [1] . Although WSNs are used in many applications, they have several limitations including limited energy supply and limited computation and communication abilities. These limitations should be considered when designing protocols for WSNs. There are two types of WSNs: structured and unstructured. An unstructured WSN is one that contains a dense collection of sensor nodes. The sensor nodes may be deployed in an ad hoc manner into the field. Once deployed, the network is left unattended to perform monitoring and reporting functions. In an unstructured WSN, network maintenance such as connection management and failures detection is difficult since there are so many nodes to take care of. In a structured WSN, all or some of the sensor nodes are deployed in a preplanned manner. The advantage of a structured network is that fewer nodes can be deployed with lower network maintenance and management cost. Fewer nodes can be deployed now since nodes are placed at specific locations to provide coverage while ad hoc deployment can have uncovered regions. The wireless sensor networks are based on the cooperation of a number of tiny sensors and which are depending upon four parts: sensor (motes), processor, transceiver, and battery. The Sensor get information from surrounding area and processor change the analog information into digital information[2]. These sensors sense and detect various environmental parameters such as temperature, pressure, air pollution etc. They are also deployed in monitoring of agriculture, smart homes, structures, passive localization, tracking etc. Then transceiver transmits the converted data to the base- station directly, or through neighboring sensor. Figure 1 shows the architecture of a typical Wireless Sensor Network with the components of a sensor node.

In this section first we discuss some of the characteristics and requirements that are sought in the design and development of a wireless sensor node. When a WSN is being implemented, particular sensor nodes features must be taken into account. These are the following: • High energy efficiency, in order to increase the node autonomy. • Low cost, as a network that covers a large area can consist of hundreds or thousands of nodes. An estimation of the number of the nodes that are required to cover a given area is presented in. • Distributed Sensing, in order to cover a large area despite the obstacles in the environment. • Wireless communication, as it is the only choice for nodes deployed in remote areas or where no cabling infrastructure is available. • Multi-hop networking. Depending on the radio parameters, it can be more efficient to reach a distant node or a base station using two or more wireless hops than a single large distance hop. • Local data processing in the node, like zero suppression, data compression and parameter extraction can reduce the transmitted payload, and, thus, the power consumption. The design of routing protocols[3] in WSNs is influenced by many challenging factors. These factors must be overcome before efficient communication can be achieved. Following some of the routing challenges and design issues that affect routing process in WSNs, are summarized. 1) Node Deployment Node deployment in WSNs is application dependent and affects the performance of the routing protocol. The deployment can be either deterministic or randomized. In deterministic deployment, the sensors are manually placed and data is routed through predetermined paths. However, in random node deployment, the sensor nodes are scattered randomly creating an infrastructure in an ad hoc manner. If the resultant distribution of nodes is not uniform, optimal clustering becomes necessary to allow connectivity and enable energy efficient network operation. Inter-sensor communication is normally within short transmission ranges due to energy and bandwidth limitations. Therefore, it is most likely that a route will consist of multiple wireless hops. 2) Energy Consumption without Losing Accuracy The sensor nodes can use up their limited supply of energy performing computations and transmitting information in a wireless environment. As such, energy conserving forms of communication and computation are essential. Sensor node lifetime shows a strong dependence on the battery lifetime. In a multihop WSN, each node plays a dual role as data sender and data router. The malfunctioning of some sensor nodes due to power failure can cause significant topological changes and might require rerouting of packets and reorganization of the network. 3) Data Reporting Model Data sensing and reporting in WSNs is dependent on the application and the time criticality of the data reporting. Data reporting can be categorized as either time- driven (continuous), event-driven, querydriven, and hybrid. The time-driven delivery model is suitable for applications that require periodic data monitoring[4]. As such, sensor nodes will periodically switch on their sensors and transmitters, sense the environment and transmit the data of interest at constant periodic time intervals.

In event-driven and query-driven models, sensor nodes react immediately to sudden and drastic changes in the value of a sensed attribute due to the occurrence of a certain event or a query is generated by the BS. As such, these are well suited for time critical applications. A combination of the previous models is also possible. The routing protocol is highly influenced by the data reporting model with regard to energy consumption and route stability. 4) Node/Link Heterogeneity In many studies, all sensor nodes are assumed to be homogeneous, i.e., having equal capacity in terms of computation, communication, and power. However, depending on the application a sensor node can have different role or capability. The existence of heterogeneous set of sensors raises many technical issues related to data routing. For example, some applications might require a diverse mixture of sensors for monitoring temperature, pressure and humidity of the surrounding environment, detecting motion via acoustic signatures, and capturing the image or video tracking of moving objects. These special sensors can be either deployed independently or the different functionalities can be included in the same sensor nodes. Even data reading and reporting can be generated from these sensors at different rates, subject to diverse quality of service constraints, and can follow multiple data reporting models. For example, hierarchical protocols designate a cluster-head node different from the normal sensors. These cluster-heads can be chosen from the deployed sensors or can be more powerful than other sensor nodes in terms of energy, bandwidth, and memory. Hence, the burden of transmission to the BS is handled by the set of cluster-heads. 5) Fault-Tolerance Some sensor nodes may fail or be blocked due to lack of power, physical damage, or environmental interference. The failure of sensor nodes should not affect the overall task of the sensor network. If many nodes fail, MAC and routing protocols must accommodate formation of new links and routes to the data collection base stations[5]. This may require actively adjusting transmit powers and signaling rates on the existing links to reduce energy consumption, or rerouting packets through regions of the network where more energy is available. Therefore, multiple levels of redundancy may be needed in a faulttolerant sensor network. 6) Scalability The number of sensor nodes deployed in the sensing area may be in the order of hundreds or thousands, or more. Any routing scheme must be able to work with this huge number of sensor nodes. In addition, sensor net-work routing protocols should be scalable enough to respond to events in the environment. Until an event occurs, most of the sensors can remain in the sleep state, with data from the few remaining sensors providing a coarse quality. 7) Network Dynamics Most of the network architectures assume that sensor nodes are stationary. However, mobility of both BSs and sensor nodes is sometimes necessary in many applications [8] . Routing messages from or to moving nodes is more challenging since route stability becomes an important issue, in addition to energy, bandwidth etc. Moreover, the sensed phenomenon can be either dynamic or static depending on the application, e.g., it is dynamic in a target detection/tracking application, while it is static in forest monitoring for early fire prevention. Monitoring static events allows the network to work in a reactive mode, simply generating traffic when reporting. Dynamic events in most applications require periodic reporting and consequently generate significant traffic to be routed to the BS. 8) Transmission Media In a multi-hop sensor network, communicating nodes are linked by a wireless medium. The traditional problems associated with a wireless channel (e.g., fading, high error rate) may also affect the operation of the sensor network. In general, the required bandwidth of sensor data will be low, on the order of 1-100 kbps. Related to the transmission media is the design of medium access control (MAC). One approach of MAC design for sensor networks is to use TDMA based protocols that conserve more energy compared to contention based protocols like CSMA (e.g., IEEE 802.11). 9) Connectivity International Journal of Science, Engineering and Technology Research (IJSETR), Volume 3, Issue 11, November 2014 3129 ISSN: 2278 – 7798 All Rights Reserved © 2014 IJSETR High node density in sensor networks precludes them from being completely isolated from each other. Therefore, sensor nodes are expected to be highly connected. This, however, may not prevent the network topology from being variable and the network size from being shrinking due to sensor node failures. In addition, connectivity depends on the possibly random distribution of nodes. 10) Coverage In WSNs, each sensor node obtains a certain view of the environment. A given sensors view of the environment is limited both in range and in accuracy; it can only cover a limited physical area of the environment. Hence, area coverage is also an important design parameter in WSNs. 11) Data Aggregation/Fusion Since sensor nodes may generate significant redundant data, similar packets from multiple nodes can be aggregated so that the number of transmissions is reduced. Data aggregation is the combination of data from different sources according to a certain aggregation function, e.g., duplicate suppression, minima, maxima and average. This technique has been used to achieve energy efficiency and data transfer optimization in a number of routing protocols. Signal processing methods can also be used for data aggregation. In this case, it is referred to as data fusion where a node is capable of producing a more accurate output signal by using some techniques such as beamforming to combine the incoming signals and reducing the noise in these signals. 12) Quality of Service In some applications, data should be delivered within a certain period of time from the moment it is sensed; otherwise the data will be useless. Therefore bounded latency for data delivery is another condition for time- constrained applications. However, in many applications, conservation of energy, which is directly related to net-work lifetime, is considered relatively more important than the quality of data sent. As the energy gets depleted, the network may be required to reduce the quality of the results in order to reduce the energy dissipation in the nodes and hence lengthen the total network lifetime. Hence, energy-aware routing protocols are required to capture this requirement.

2. What is meant by OLSR and explain about OLSR routing protocol [CO5-L2]

Features specific to OLSR[

Link-state routing protocols such as Open Shortest Path First (OSPF) and IS-IS elect a *designated router* on every link to perform flooding of topology information. In wireless ad hoc networks, there is different notion of a link, packets can and do go out the same interface; hence, a different approach is needed in order to optimize the flooding process. Using Hello messages the OLSR protocol at each node discovers 2-hop neighbor information and performs a distributed election of a set of *multipoint relays* (MPRs). Nodes select MPRs such that there exists a path to each of its 2-hop neighbors via a node selected as an MPR. These MPR nodes then source and forward TC messages that contain the MPR selectors. This functioning of MPRs makes OLSR unique from other link state routing protocols in a few different ways: The forwarding path for TC messages is not shared among all nodes but varies depending on the source, only a subset of nodes source link state information, not all links of a node are advertised but only those that represent MPR selections.

Since link-state routing requires the topology database to be synchronized across the network, OSPF and IS-IS perform topology flooding using a reliable algorithm. Such an algorithm is very difficult to design for ad hoc wireless networks, so OLSR doesn't bother with reliability; it simply floods topology data often enough to make sure that the database does not remain unsynchronized for extended periods of time.

Multipoint relays

Multipoint relays (MPRs) relay messages between nodes. They also have the main role in routing and selecting the proper route from any source to any desired destination node.

MPRs advertise link-state information for their MPR selectors (a node selected as a MPR) periodically in their control messages. MPRs are also used to form a route from a given node to any destination in route calculation. Each node periodically broadcasts a Hello message for the link sensing, neighbor detection and MPR selection processes.

Benefits

Being a proactive protocol, routes to all destinations within the network are known and maintained before use. Having the routes available within the standard routing table can be useful for some systems and network applications as there is no route discovery delay associated with finding a new route.

The routing overhead generated, while generally greater than that of a reactive protocol, does not increase with the number of routes being created.

Default and network routes can be injected into the system by HNA messages allowing for connection to the internet or other networks within the OLSR MANET cloud. Network routes are something reactive protocols do not currently execute well.

Timeout values and validity information is contained within the messages conveying information allowing for differing timer values to be used at differing nodes.

Criticisms

The original definition of OLSR does not include any provisions for sensing of link quality; it simply assumes that a link is up if a number of hello packets have been received recently. This assumes that links are bi-modal (either working or failed), which is not necessarily the case on wireless networks, where links often exhibit intermediate rates of packet loss. Implementations such as the open source OLSRd (commonly used on Linux-based mesh routers) have been extended (as of v. 0.4.8) with link quality sensing.

Being a proactive protocol, OLSR uses power and network resources in order to propagate data about possibly unused routes. While this is not a problem for wired access points, and laptops, it makes OLSR unsuitable for sensor networks that try to sleep most of the time. For small scale wired access points with low CPU power, the open source OLSRd project showed that large scale mesh networks can run with OLSRd on thousands of nodes with very little CPU power on 200 MHz embedded devices.

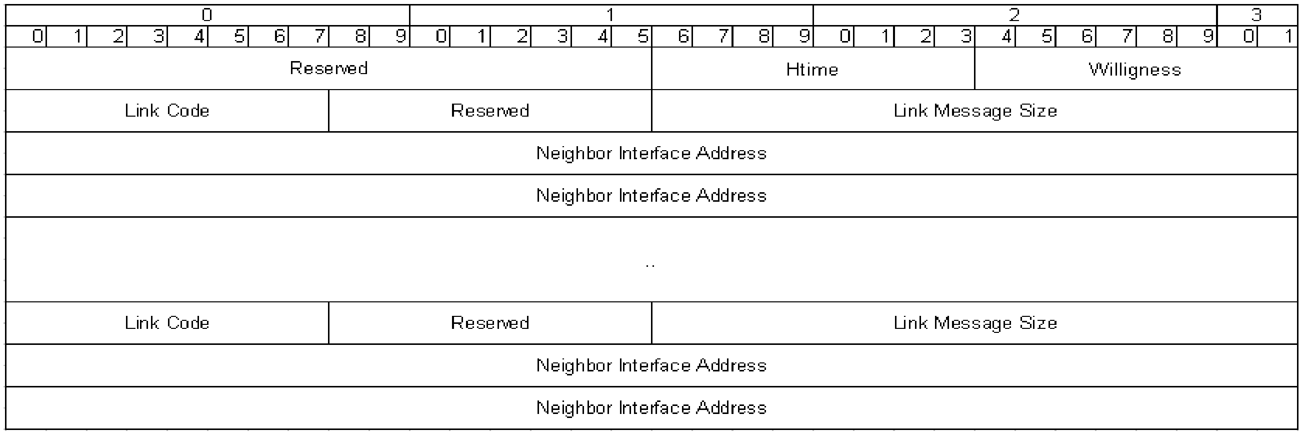
Being a link-state protocol, OLSR requires a reasonably large amount of bandwidth and CPU power to compute optimal paths in the network. In the typical networks where OLSR is used (which rarely exceed a few hundreds of nodes), this does not appear to be a problem.

By only using MPRs to flood topology information, OLSR removes some of the redundancy of the flooding process, which may be a problem in networks with moderate to large packet loss rates[3] – however the MPR mechanism is self-pruning (which means that in case of packet losses, some nodes that would not have retransmitted a packet, may do so).

Messages

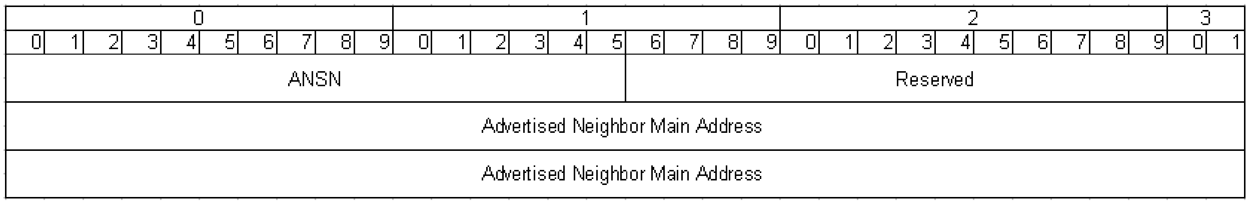
OLSR makes use of "Hello" messages to find its one hop neighbors and its two hop neighbors through their responses. The sender can then select its multipoint relays (MPR) based on the one hop node that offers the best routes to the two hop nodes. Each node has also an MPR selector set, which enumerates nodes that have selected it as an MPR node. OLSR uses topology control (TC) messages along with MPR forwarding to disseminate neighbor information throughout the network. *Host and network association* (HNA) messages are used by OLSR to disseminate network route advertisements in the same way TC messages advertise host routes.

Hello



<table>
<tr><td colspan="10">0</td><td colspan="10">1</td><td colspan="10">2</td><td colspan="2">3</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="16">Reserved</td><td colspan="8">Htime</td><td colspan="8">Willigness</td></tr>
<tr><td colspan="8">Link Code</td><td colspan="8">Reserved</td><td colspan="16">Link Message Size</td></tr>
<tr><td colspan="32">Neighbor Interface Address</td></tr>
<tr><td colspan="32">Neighbor Interface Address</td></tr>
<tr><td colspan="32">..</td></tr>
<tr><td colspan="8">Link Code</td><td colspan="8">Reserved</td><td colspan="16">Link Message Size</td></tr>
<tr><td colspan="32">Neighbor Interface Address</td></tr>
<tr><td colspan="32">Neighbor Interface Address</td></tr>
</table>

Topology control (TC)



<table>
<tr><td colspan="10">0</td><td colspan="10">1</td><td colspan="10">2</td><td colspan="2">3</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="16">ANSN</td><td colspan="16">Reserved</td></tr>
<tr><td colspan="32">Advertised Neighbor Main Address</td></tr>
<tr><td colspan="32">Advertised Neighbor Main Address</td></tr>
</table>

Other approaches

The problem of routing in ad hoc wireless networks is actively being researched, and OLSR is but one of several proposed solutions. To many, it is not clear whether a whole new protocol is needed, or whether OSPF could be extended with support for wireless interfaces.[4][5]

In bandwidth- and power-starved environments, it is interesting to keep the network silent when there is no traffic to be routed. Reactive routing protocols do not maintain routes, but build them on demand. As link-state protocols require database synchronisation, such protocols typically use the distance vector approach, as in AODV and DSDV, or more ad hoc approaches that do not necessarily build optimal paths, such as Dynamic Source Routing.

For more information see the list of ad hoc routing protocols.

OLSR version 2

OLSRv2 has been published by the IETF in April 2014.[6] It maintains many of the key features of the original including MPR selection and dissemination. Key differences are the flexibility and modular design using shared components: packet format packetbb, and neighborhood discovery protocol NHDP. These components are being designed to be common among next generation IETF MANET protocols. Differences in the handling of multiple address and interface enabled nodes is also present between OLSR and OLSRv2.

Implementations

- OLSR.ORG – Downloadable code for OLSR on GNU/Linux, Windows, Mac OS X, FreeBSD and NetBSD systems. Features a great deal of documentation, including an informative survey of related work.
- NRL-OLSR – Open source code of NRL-OLSR. Works on Windows, MacOS, Linux, and various embedded PDA systems such as Arm/Zaurus and PocketPC as well as simulation environments ns2 and OPNET., http://cs.itd.nrl.navy.mil/focus/
- SOURCEFORGE.NET-OLSR – Created by MOVIQUITY and based on studies within the project Workpad, it offers a code in C# to deploy a MANET (Ad Hoc, Meshnet) with protocol OLSR. Developed for WM 6, Win XP and can be adapted to other platforms using .Net Framework and Compact

3. Explain about the absolute and relative localization [CO5-L1]

An indoor positioning system (IPS) is a system to locate objects or people inside a building using radio waves, magnetic fields, acoustic signals, or other sensory information collected by mobile devices.[1] There are several commercial systems on the market, but there is no standard for an IPS system.

IPSes use different technologies, including distance measurement to nearby anchor nodes (nodes with known positions, e.g., WiFi access points), magnetic positioning, dead reckoning. They either actively locate mobile devices and tags or provide ambient location or environmental context for devices to get sensed.[2]

The localized nature of an IPS has resulted in design fragmentation, with systems making use of various optical,[3] radio,[4][5][6][7][8] or even acoustic[9] technologies.

System designs must take into account that at least three independent measurements are needed to unambiguously find a location (see trilateration). For smoothing to compensate for stochastic (unpredictable) errors there must be a sound method for reducing the error budget significantly. The system might include information from other systems to cope for physical ambiguity and to enable error compensation.

Applicability and precision

Due to the signal attenuation caused by construction materials, the satellite based Global Positioning System (GPS) loses significant power indoors affecting the required coverage for receivers by at least four satellites. In addition, the multiple reflections at surfaces cause multi-path propagation serving for uncontrollable errors. These very same effects are degrading all known solutions for indoor locating which uses electromagnetic waves from indoor transmitters to indoor receivers. A bundle of physical and mathematical methods is applied to compensate for these problems. Promising direction radiofrequency positioning error correction opened by the use of alternative sources of navigational information, such as inertial measurement unit (IMU), monocular camera Simultaneous localization and mapping (SLAM) and WiFi SLAM. Integration of data from various navigation systems with different physical principles can increase the accuracy and robustness of the overall solution

Relation to GPS

Global navigation satellite systems (GPS or GNSS) are generally not suitable to establish indoor locations, since microwaves will be attenuated and scattered by roofs, walls and other objects. However, in order to make positioning signals ubiquitous, integration between GPS and indoor positioning can be made.

Currently, GNSS receivers are becoming more and more sensitive due to ceaseless progress in chip technology and processing power. High Sensitivity GNSS receivers are able to receive satellite signals in most indoor environments and attempts to determine the 3D position indoors have been successful.[20] Besides increasing the sensitivity of the receivers, the technique of A-GPS is used, where the almanac and other information are transferred through a mobile phone.

However, proper coverage for the required four satellites to locate a receiver is not achieved with all current designs (2008–11) for indoor operations. Beyond, the average error budget for GNSS systems normally is much larger than the confinements, in which the locating shall be performed.

Locating and positioning

While most current IPS systems are able to detect the location of an object, they are so coarse that they cannot be used to detect the *orientation* or *direction* of an object.

Locating and tracking

One of the methods to thrive for sufficient operational suitability is "tracking". Whether a sequence of locations determined form a trajectory from the first to the most actual location. Statistical methods then serve for smoothing the locations determined in a track resembling the physical capabilities of the object to move. This smoothing must be applied, when a target moves and also for a resident target, to compensate erratic measures. Otherwise the single resident location or even the followed trajectory would compose of an itinerant sequence of jumps.

Identification and segregation

In most applications the population of targets is larger than just one. Hence the IPS must serve a proper specific identification for each observed target and must be capable to segregate and separate the targets individually within the group. An IPS must be able to identify the entities being tracked, despite the "non-interesting" neighbors. Depending on the design, either a sensor network must know from which tag it has received information, or a locating device must be able to identify the targets directly.

Non-radio technologies

Non-radio technologies can be used for positioning without using the existing wireless infrastructure. This can provide increased accuracy at the expense of costly equipment and installations.

Magnetic positioning

Magnetic positioning can offer pedestrians with smartphones an indoor accuracy of 1–2 meters with 90% confidence level, without using the additional wireless infrastructure for positioning. Magnetic positioning is based on the iron inside buildings that create local variations in the Earth's magnetic field. Un-optimized compass chips inside smartphones can sense and record these magnetic variations to map indoor locations

Inertial measurements

Pedestrian dead reckoning and other approaches for positioning of pedestrians propose an inertial measurement unit carried by the pedestrian either by measuring steps indirectly (step counting) or in a foot mounted approach,[23] sometimes referring to maps or other additional sensors to constrain the inherent sensor drift encountered with inertial navigation. However, in order to make it capable to build map itself, the SLAM algorithm framework [24] will be used.

Inertial measures generally cover the differentials of motion, hence the location gets determined with integrating and thus requires integration constants to provide results.[28][29]The actual position estimation can be found as the maximum of a 2-d probability distribution which is recomputed at each step taking into account the noise model of all the sensors involved and the constraints posed by walls and furniture

Wireless technologies

opology options for hardware and software configuration, network-based, terminal-based, and terminal-assisted. Positioning accuracy can be increased at the expense of wireless infrastructure equipment and installations.

Wi-Fi-based positioning system (WPS)

Wi-Fi positioning system (WPS) is used where GPS is inadequate. The localization technique used for positioning with wireless access points is based on measuring the intensity of the received signal (*received signal strength* in English RSS) and the method of "fingerprinting".[31][32][33] Typical parameters useful to geolocate the WiFi hotspot or wireless access point include the SSID and the MAC address of the access point. The accuracy depends on the number of positions that have been entered into the database. The possible signal fluctuations that may occur can increase errors and inaccuracies in the path of the user. Anyplace[34] is a free and open-source Wi-Fi positioning system that allows anybody to rapidly map indoor spaces and that won several awards for its location accuracy

Bluetooth

According to the Bluetooth Special Interest Group,[37] Bluetooth is all about proximity, not about exact location. Bluetooth was not intended to offer a pinned location like GPS, however is known as a geo-fence or micro-fence solution which makes it an indoor proximity solution, not an indoor positioning solution. Micromapping and indoor mapping[38]has been linked to Bluetooth[39] and to the Bluetooth LE based iBeacon promoted by Apple Inc.. Large-scale indoor positioning system based on iBeacons has been implemented and applied in practice

Choke point concepts

Simple concept of location indexing and presence reporting for tagged objects, uses known sensor identification only.[7] This is usually the case with passive radio-frequency identification (RFID) systems, which do not report the signal strengths and various distances of single tags or of a bulk of tags and do not renew any before known location coordinates of the sensor or current location of any tags. Operability of such approaches requires some narrow passage to prevent from passing by out of range.

Grid concepts

Instead of long range measurement, a dense network of low-range receivers may be arranged, e.g. in a grid pattern for economy, throughout the space being observed. Due to the low range, a tagged entity will be identified by only a few close, networked receivers. An identified tag must be within range of the identifying reader, allowing a rough approximation of the tag location. Advanced systems combine visual coverage with a camera grid with the wireless coverage for the rough location.

Long range sensor concepts

Most systems use a continuous physical measurement (such as angle and distance or distance only) along with the identification data in one combined signal. Reach by these sensors mostly covers an entire floor, or an aisle or just a single room. Short reach solutions get applied with multiple sensors and overlapping reach.

Angle of arrival

Angle of arrival (AoA) is the angle from which a signal arrives at a receiver. AoA is usually determined by measuring the time difference of arrival (TDOA) between multiple antennas in a sensor array. In other receivers, it is determined by an array of highly directional sensors—the angle can be determined by which sensor received the signal. AoA is usually used with triangulation and a known base line to find the location relative to two anchor transmitters.

Time of arrival

Time of arrival (ToA, also time of flight) is the amount of time a signal takes to propagate from transmitter to receiver. Because the signal propagation rate is constant and known (ignoring differences in mediums) the travel time of a signal can be used to directly calculate distance. Multiple measurements can be combined with trilateration and multilateration to find a location. This is the technique used by GPS. Systems which use ToA, generally require a complicated synchronization mechanism to maintain a reliable source of time for sensors (though this can be avoided in carefully designed systems by using repeaters to establish coupling.

The accuracy of the TOA based methods often suffers from massive multipath conditions in indoor localization, which is caused by the reflection and diffraction of the

RF signal from objects (e.g., interior wall, doors or furniture) in the environment. However, it is possible to reduce the effect of multipath by applying temporal or spatial sparsity based techniques.[42] [43]

Received signal strength indication

Received signal strength indication (RSSI) is a measurement of the power level received by sensor. Because radio waves propagate according to the inverse-square law, distance can be approximated based on the relationship between transmitted and received signal strength (the transmission strength is a constant based on the equipment being used), as long as no other errors contribute to faulty results. The inside of buildings is not free space, so accuracy is significantly impacted by reflection and absorption from walls. Non-stationary objects such as doors, furniture, and people can pose an even greater problem, as they can affect the signal strength in dynamic, unpredictable ways.

A lot of systems use enhanced Wi-Fi infrastructure to provide location information.[4][5][6] None of these systems serves for proper operation with any infrastructure as is. Unfortunately, Wi-Fi signal strength measurements are extremely noisy, so there is ongoing research focused on making more accurate systems by using statistics to filter out the inaccurate input data. Wi-Fi Positioning Systems are sometimes used outdoors as a supplement to GPS on mobile devices, where only few erratic reflections disturb the results.

4. Write notes on triangulation [CO5-L2]

Specifically in surveying, triangulation per se involves only angle measurements, rather than measuring distances to the point directly as in trilateration; the use of both angles and distance measurements is referred to as triangulateration.

Applications

Optical 3D measuring systems use this principle as well in order to determine the spatial dimensions and the geometry of an item. Basically, the configuration consists of two sensors observing the item. One of the sensors is typically a digital camera device, and the other one can also be a camera or a light projector. The projection centers of the sensors and the considered point on the object’s surface define a (spatial) triangle. Within this triangle, the distance between the sensors is the base *b* and must be known. By determining the angles between the projection rays of the sensors and the basis, the intersection point, and thus the 3D coordinate, is calculated from the triangular relations.

History

The use of triangles to estimate distances dates to antiquity. In the 6th century BC, just prior to the establishment of the Ptolemaic dynasty, the Greek philosopher Thales is recorded as using similar triangles to estimate the height of the pyramids of ancient Egypt. He measured the length of the pyramids' shadows and that of his own at the same moment, and compared the ratios to his height (intercept theorem).[1] Thales also estimated the distances to ships at sea as seen from a clifftop by measuring the horizontal distance traversed by the line-of-sight for a known fall, and scaling up to the height of the whole cliff.[2] Such techniques would have been familiar to the ancient Egyptians. Problem 57 of the Rhind papyrus, a thousand years earlier, defines the *seqt* or *seked* as the ratio of the run to the rise of a slope, *i.e.* the reciprocal of gradients as measured today. The slopes and angles were measured using a sighting rod that the Greeks called a *dioptra*, the forerunner of the Arabic alidade. A detailed contemporary collection of constructions for the determination of lengths from a distance using this instrument is known, the *Dioptra* of Hero of Alexandria (c. 10–70 AD), which survived in Arabic translation; but the knowledge became lost in Europe. In China, Pei Xiu (224–271) identified "measuring right angles and acute angles" as the fifth of his six principles for accurate map-making, necessary to accurately establish distances;[3] while Liu Hui (c. 263) gives a version of the calculation above, for measuring perpendicular distances to inaccessible places

Mobile phone tracking is the ascertaining of the position or location of a mobile phone, whether stationary or moving. Localization may occur either via multilateration of radio signals between (several) cell towers of the network and the phone, or simply via GPS. To locate a mobile phone using multilateration of radio signals, it must emit at least the roaming signal to contact the next nearby antenna tower, but the process does not require an active call. The Global System for Mobile Communications (GSM) is based on the phone's signal strength to nearby antenna masts.[1]

Mobile positioning may include location-based services that disclose the actual coordinates of a mobile phone, which is a technology used by telecommunication companies to approximate the location of a mobile phone, and thereby also its user.

Technology

The technology of locating is based on measuring power levels and antenna patterns and uses the concept that a powered mobile phone always communicates wirelessly with one of the closest base stations, so knowledge of the location of the base station implies the cell phone is nearby.

Advanced systems determine the sector in which the mobile phone is located and roughly estimate also the distance to the base station. Further approximation can be done by interpolating signals between adjacent antenna towers. Qualified services may achieve a precision of down to 50 meters in urban areas where mobile traffic and density of antenna towers (base stations) is sufficiently high.Rural and desolate areas may see miles between base stations and therefore determine locations less precisely.

GSM localization uses multilateration to determine the location of GSM mobile phones, or dedicated trackers, usually with the intent to locate the user.

The location of a mobile phone can be determined in a number of ways:

Network-based

The location of a mobile phone can be determined using the service provider's network infrastructure. The advantage of network-based techniques, from a service provider's point of view, is that they can be implemented non-intrusively without affecting handsets. Network-based techniques were developed many years prior to the widespread availability of GPS on handsets. (See US 5519760, issued 21 May 1996 for one of the first works relating to this.[3])

The accuracy of network-based techniques varies, with cell identification as the least accurate and triangulation as moderately accurate, and newer "advanced forward link trilateration" timing methods as the most accurate. The accuracy of network-based techniques is both dependent on the concentration of cell base stations, with urban environments achieving the highest possible accuracy because of the higher number of cell towers, and the implementation of the most current timing methods.

One of the key challenges of network-based techniques is the requirement to work closely with the service provider, as it entails the installation of hardware and software within the operator's infrastructure. Frequently the compulsion associated with a legislative framework, such as Enhanced 9-1-1, is required before a service provider will deploy a solution.

Handset-based

The location of a mobile phone can be determined using client software installed on the handset.[4] This technique determines the location of the handset by putting its location by cell identification, signal strengths of the home and neighboring cells, which is continuously sent to the carrier. In addition, if the handset is also equipped with GPS then significantly more precise location information can be then sent from the handset to the carrier.

Another approach is to use a fingerprinting-based technique,[5][6][7] where the "signature" of the home and neighboring cells signal strengths at different points in the area of interest is recorded by war-driving and matched in real-time to determine the handset location. This is usually performed independent from the carrier.

The key disadvantage of handset-based techniques, from service provider's point of view, is the necessity of installing software on the handset. It requires the active cooperation of the mobile subscriber as well as software that must be able to handle the different operating systems of the handsets. Typically, smartphones, such as one based on Symbian, Windows Mobile, Windows Phone, BlackBerry OS, iOS, or Android, would be able to run such software, e.g. Google Maps.

One proposed work-around is the installation of embedded hardware or software on the handset by the manufacturers, e.g., Enhanced Observed Time Difference (E-OTD). This avenue has not made significant headway, due to the difficulty of convincing different manufacturers to cooperate on a common mechanism and to address the cost issue. Another difficulty would be to address the issue of foreign handsets that are roaming in the network.

SIM-based

Using the subscriber identity module (SIM) in GSM and Universal Mobile Telecommunications System (UMTS) handsets, it is possible to obtain raw radio measurements from the handset.[8][9] Available measurements include the serving Cell ID, round-trip time, and signal strength. The type of information obtained via the SIM can differ from that which is available from the handset. For example, it may not be possible to obtain any raw measurements from the handset directly, yet still obtain measurements via the SIM.

Wi-Fi

Crowdsourced Wi-Fi data can also be used to identify a handset's location.[10] Poor performance of the GPS-based methods in indoor environment and increasing popularity of Wi-Fi have encouraged companies to design new and feasible methods to carry out Wi-Fi-based indoor positioning.[11] Most smartphones combine Global Navigation Satellite Systems (GNSS), such as GPS and GLONASS, with Wi-Fi positioning systems.

Hybrid

Hybrid positioning systems use a combination of network-based and handset-based technologies for location determination. One example would be some modes of Assisted GPS, which can both use GPS and network information to compute the location. Both types of data are thus used by the telephone to make the location more accurate (i.e., A-GPS). Alternatively tracking with both systems can also occur by having the phone attain its GPS-location directly from the satellites, and then having the information sent via the network to the person that is trying to locate the telephone. Such systems include Google Maps, as well as, LTE's OTDOA and E-CellID.

There are also hybrid positioning systems which combine several different location approaches to position mobile devices by Wi-Fi, WiMAX, GSM, LTE, IP addresses, and network environment data.

Operational purpose

In order to route calls to a phone, the cell towers listen for a signal sent from the phone and negotiate which tower is best able to communicate with the phone. As the phone changes location, the antenna towers monitor the signal, and the phone is "roamed" to an adjacent tower as appropriate. By comparing the relative signal strength from multiple antenna towers, a general location of a phone can be roughly determined. Other means make use of the antenna pattern, which supports angular determination and phase discrimination.

Newer phones may also allow the tracking of the phone even when turned on and not active in a telephone call. This results from the roaming procedures that perform hand-over of the phone from one base station to another.[12]

Bearer interest.

A phone's location can be uploaded to a common website where one's friends and family can view one's last reported position. Newer phones may have built-in GPS receivers which could be used in a similar fashion, but with much higher accuracy. This is controversial, because data on a common website means people who are not "friends and family" may be able to view the information.

Privacy

Locating or positioning touches upon delicate privacy issues, since it enables someone to check where a person is without the person's consent. Strict ethics and security measures are strongly recommended for services that employ positioning. In 2012 Malte Spitz held a TED talk[13] on the issue of mobile phone privacy in which he showcased his own stored data that he received from Deutsche Telekom after suing the company. He described the data, which consists of 35,830 lines of data collected during the span of Germany's data retention at the time, saying, "This is six months of my life [...] You can see where I am, when I sleep at night, what I'm doing." He partnered up with ZEIT Online and made his information publicly available in an interactive map which allows users to watch his entire movements during that time in fast-forward. Spitz concluded that technology consumers are the key to challenging privacy norms in today's society who "have to fight for self determination in the digital age

5. Explain about the QOS in WSN [CO5-L2]

Quality of service is an overused term with multiple meanings and perspectives from different research and technical communities [1]. QoS in WSNs can be viewed from two perspectives: application-specific and network. The former refers to QoS parameters specific to the application, such as sensor node measurement, deployment, and coverage and number of active sensor nodes. The latter refers to how the supporting communication network can meet application needs while efficiently using network resources such as bandwidth and power consumption.

With the recent technological developments of the wireless networks and multifunctional sensors with processing and communication capabilities, wireless sensor networks (WSNs) have been used in an increasing number of applications. WSNs can provide a more accurate or reliable monitoring service for different classes of applications [2,3]. Quality of service can be an important mechanism to guarantee that the distinct requirements for different classes of applications are met [4].

Traditional QoS mechanisms used in wired networks aren't adequate for WSNs because of constraints such as resource limitations and dynamic topology. One of the many challenges concerning wireless sensor networks (WSNs) is how to provide Quality of Service (QoS) parameter guarantees in real-time applications [5]. Therefore, middleware should provide new mechanisms to maintain QoS over an extended period and even adjust itself when the required QoS and the state of the application changes. Middleware should be designed based on trade-offs among performance metrics such as network capacity or throughput, data delivery delay, and energy consumption in order to provide QoS in Wireless Sensor Network.

QoS Concept

As defined in [6], Quality-of-Service is a set of service requirements to be met by the network while transporting a flow. "Here a flow is" a packet stream from source to a destination (unicast or multicast) with an associated Quality of Service (QoS) [6]. In other words, QoS is a measurable level of service delivered to network users, which can be characterized by packet loss probability, available bandwidth, end-to-end delay, etc. Such QoS can be provided by network service providers in terms of some agreement (Service Level Agreement, or SLA) between network users and service providers. For example, users can require that for some traffic flows, the network should choose a path with minimum 2M bandwidth.

QoS Metrics

For quality of service to be implemented, service requirements have to be expressed in some measurable QoS metrics. The well-known metrics include bandwidth, delay, jitter, cost, loss probability, etc. Different metrics may have different features. There are 3 types of metrics when talking about QoS: additive, multiplicative, and concave [7]. These can be defined as follows:

Let m (n_1, n_2) be a metric for link (n_1, n_2). For any path $P = (n_1, n_2, \cdots, n_i, n_j)$, metric m is: (Note here $n_1, n_2, n_3, \cdots, n_i, n_j$ represent network nodes)

additive, if $m(P) = m(n_1, n_2) + m(n_2, n_3) + \cdots + m(n_i, n_j)$

Examples are delay, jitter, cost and hop-count. For instance, the delay of a path is the sum of the delay of every hop.

multiplicative, if $m(P) = m(n_1, n_2) * m(n_2, n_3) * \cdots * m(n_i, n_j)$

Example is reliability, in which case $0 < m(n_i, n_j) < 1$.

concave, if $m(P) = \min\{m(n_1, n_2), m(n_2, n_3), \cdots, m(n_i, n_j)\}$

Example is bandwidth, which means that the bandwidth of a path is determined by the link with the minimum available bandwidth.

QoS Challenges in Sensor Networks

Different from IP network, Sensor network naturally supports multiple service types, thus provides different QoS. The service types range from CBR (Constant Bit Rate) which guarantees bandwidth, delay and delay jitter, to UBR (Unspecified Bit Rate) which virtually provides no guarantees (just like today's “best-effort” IP network). While sensor networks inherit most of the QoS issues from the general wireless networks, their characteristics pose unique challenges. The following is an outline of design considerations for handling QoS traffic in wireless sensor networks.

Bandwidth limitation: A typical issue for general wireless networks is securing the bandwidth needed for achieving the required QoS. Bandwidth limitation is going to be a more pressing issue for wireless sensor networks. Traffic in sensor networks can be burst with a mixture of real-time and non-real-time traffic. Dedicating available bandwidth solely to QoS traffic will not be acceptable. A trade-off in image/video quality may be necessary to accommodate non-real-time traffic. In addition, simultaneously using multiple independent routes will be sometime needed to split the traffic and allow for meeting the QoS requirements. Setting up independent routes for the same flow can be very complex and challenging in sensor networks due energy constraints, limited computational resources and potential increase in collisions among the transmission of sensors.

Removal of redundancy: Sensor networks are characterized with high redundancy in the generated data. For unconstrained traffic, elimination of redundant data messages is somewhat easy since simple aggregation functions would suffice. However, conducting data aggregation for QoS traffic is much more complex. Comparison of images and video streams is not computationally trivial and can consume significant energy resources. A combination of system and sensor level rules would be necessary to make aggregation of QoS data computationally feasible. For example, data aggregation of imaging data can be selectively performed for traffic generated by sensors pointing to same direction since the images may be very similar. Another factor of consideration is the amount of QoS traffic at a particular moment. For low traffic it may be more efficient to cease data aggregation since the overhead would become dominant. Despite the complexity of data aggregation of imaging and video data, it can be very rewarding from a network performance point-of-view given the size of the data and the frequency of the transmission.

Energy and delay trade-off: Since the transmission power of radio is proportional to the distance squared or even higher order in noisy environments or in the nonflat terrain, the use of multi-hop routing is almost a standard in wireless sensor networks. Although the increase in the number of hops dramatically reduces the energy consumed for data collection, the accumulative packet delay magnifies. Since packet queuing delay dominates its propagation delay, the increase in the number of hops can, not only slow down packet delivery but also complicate the analysis and the handling of delay-constrained traffic. Therefore, it is expected that QoS routing of sensor data would have to sacrifice energy efficiency to meet delivery requirements. In addition, redundant routing of data may be unavoidable to cope with the typical high error rate in wireless communication, further complicating the trade-off between energy consumption and delay of packet delivery.

Buffer size limitation: Sensor nodes are usually constrained in processing and storage capabilities. Multi-hop routing relies on intermediate relaying nodes for storing incoming packets for forwarding to the next hop. While a small buffer size can conceivably suffice, buffering of multiple packets has some advantages in wireless sensor networks. First, the transition of the radio circuitry between transmission and reception modes consumes considerable energy and thus it is advantageous to receive many packets prior to forwarding them. In addition, data aggregation and fusion involves multiple packets. Multihop routing of QoS data would typically require long sessions and buffering of even larger data, especially when the delay jitter is of interest. The buffer size limitation will increase the delay variation that packets incur while traveling on different routes and even on the same route. Such an issue will complicate medium access scheduling and make it difficult to meet QoS requirements.

Support of multiple traffic types: Inclusion of heterogeneous set of sensors raises multiple technical issues related to data routing. For instance, some applications might require a diverse mixture of sensors for monitoring temperature, pressure and humidity of the surrounding environment, detecting motion via acoustic signatures and capturing the image or video tracking of moving objects. These special sensors are either deployed independently or the functionality can be included on the normal sensors to be used on demand. Reading generated from these sensors can be at different rates, subject to diverse quality of service constraints and following multiple data delivery models, as explained earlier. Therefore, such a heterogeneous environment makes data routing more challenging.

Reliability, Availability and Serviceability

As Wireless Sensor Networks (WSNs) are expected to be adopted in many industrial, health care and military applications, their reliability, availability and serviceability (RAS) are becoming critical. In recent years, the diverse potential applications for wireless sensor networks (WSN) have been touted by researchers and the general press [8-10]. In many WSNs systems, to provide sufficient RAS can often be absorbed in the network cost. Nevertheless, as noticed early [11], network designers face “two fundamentally conflicting goals: to minimize the total cost of the network and to provide redundancy as a protection against major service interruptions.”

For availability and serviceability, remote testing and diagnostics is needed to pinpoint and repair (or bypass) the failed components that might be physically unreachable. Severe limitations in the cost and the transmitted energy within WSNs negatively impact the reliability of the nodes and the integrity of transmitted data. The application itself will greatly influence how system resources (namely, energy and bandwidth) must be allocated between communication and computation requirements to achieve requisite system performance. The presentation below demonstrates how different application wireless sensor nodes can influence the resource usability:

Power states are states of particular devices; as such, they are generally not visible to the user. For example, some devices may be in the Off state even though the system as a whole is in the working state.

These states are defined very generically in this section to enable applications adopted in our approach. Many devices do not have all four power states defined. Devices may be capable of several different low power modes, but if there is no user-perceptible difference between the modes only the lowest power mode will be used. We define four power states according to advanced configuration power interface (ACPI) [12]:

Ready—(or busy) is when the system or device is fully powered up and ready for use.

Idle—is an intermediate system dependent state that attempts to conserve power. The CPU enters the idle state when no device activity has occurred within a machine defined time period. The machine won’t return to busy state until a device raises a hardware interrupt or the machine accesses a controlled device.

Suspend—is the lowest level of power consumptions available in which memory preserves all data and operational parameters. The device won’t perform any computations until it resumes normal activity, which it does when signal by an external event such as a button press, timer alarm, or receipt of request.

When Off—the device is powered down and inactive. Operational and data parameters might or might not be preserved in

Figure 1 shows the general current ranges for each

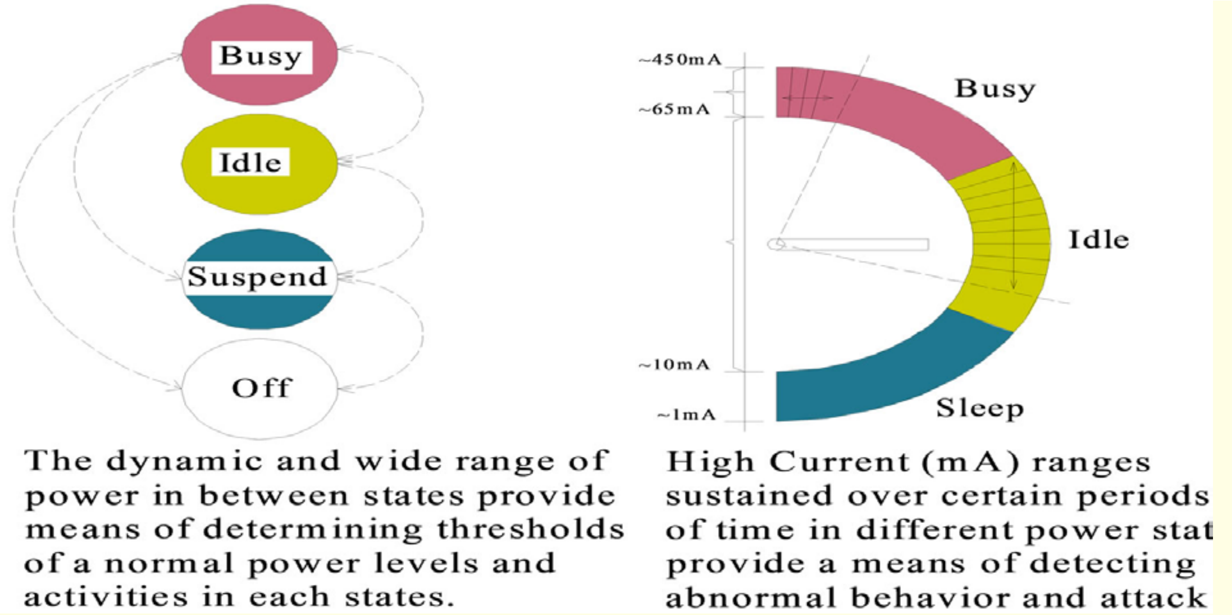

Busy
Idle
Suspend
Off
~450mA
~65mA
~10mA
~1mA
Busy
Idle
Sleep
The dynamic and wide range of power in between states provide means of determining thresholds of a normal power levels and activities in each states.
High Current (mA) ranges sustained over certain periods of time in different power stat provide a means of detecting abnormal behavior and attack

Figure 1. State power distribution (adapted from a Dell Axim) and battery-based intrusion detections (B-BID) power drain rate thresholds. The longer a threshold is held high in the busy and idle states, the greater the likehood that an anomalous activity is present.

operating state as well as the power distribution for a PDA class of devices. Cliff Brake affirms that the CPU accounts for approximately 30 percent of power and the screen 42 percent when backlit these percentages vary slightly with each PDA class [13]. In an idle state, the CPU loses nearly all current and the backlight is turned off, equating to an approximate 64 percent power reduction.

This can be deceiving, however. In idle state, if the wireless local area network (LAN) card picks up a network request and transmits an acknowledgement, the CPU will consume power at a higher level. Worse yet, once on, the card might pick up multiple requests, and unless the user has altered the CPU's communication protocol, it will try to send multiple acknowledgements for each request. In addition, the power required to transmit is greater than it is to receive by approximately 1.5:1 [14,15].

The justification of idle state resource consumption can be only identified through worse or best scenarios as follow:

$$E = N\left(1 + r/t\right)^{T} \quad (1)$$

The inputs are the total number of nodes (N), threshold (r), the time taken (t), and total time (T). One of the purposes of a model such as this is to make predictions and try "What If?" scenarios. You can change the inputs and recalculate the model and you'll get a new answer. You might even want to plot a graph of the expected results (E) vs. time (T). In some cases, you may have a fixed results rate, but what do you do if the results rate is allowed to change? For this simple equation, you might only care to know a worst/best case scenario, where you calculate the expected value based upon the lowest and highest results rates that you might expect.

While examining WSN nodes and propose the necessary QoS required for increasing both the availability and serviceability of the system our approach is service oriented and was particularly motivated by recent proposals to define QoS (quality of service) for WSN. In one definition, QoS measures application reliability with a goal of energy efficiency [16]. An alternative definition equates QoS to spatial resolution [17]. This latter work also presented a QoS control strategy based on a Gur game paradigm in which base stations broadcast feedback to the network's sensors. QoS control is required for the assumption is that the number of sensors deployed exceeds the minimum needed to provide the requisite service.

This work presents two new techniques to maintain QoS under a variety of network constraints. We first adapt the proposed Gur game strategy to operate in energy poor environments then proposes a new, extremely low-energy control strategy based on individual feedback in a random access communication system. In particular, our work is applicable to networks that are deployed in remote, harsh environs (e.g., space applications). Such networks are constrained by (1) high die-off rates of nodes and (2) inability to be replenished. The performance of the proposed algorithms is demonstrated throughout using numerical examples as follows (2) and (3):

Where m is a number of failed nodes within WSN.

n is number of nodes within WSN and $M\%$ is possible percentage of failed nodes within given WSN.

4. Calculating Probability of Nodes Availability in WSN

The availability of several implementations is derived from Equation (3) above for Mean Time between Failure (MTBF) and Mean Time to Repair (MTTR). Due to the power issue and the unpredictable wireless network characteristics, it is possible that applications running on the sensor nodes might fail. Thus, techniques to improve the availability of sensor nodes are necessary. Estimated MTBF in our sensor nodes is based on the individually calculated failure rates for each component and the circuit board. Next, for the redundant system versions, if the failure rates (λ) of each redundant element are the same, then the MTBF of the redundant system with n parallel independent elements (i) [18] are taken as:

The MTTR can be estimated by the sum of two values, referred to as Mean Time to Detect (MTTD) the failures and the Time to Repair (TTR) (MTTR = MTTD + TTR). Notice that this part might be severely affected by the network connections.

Considering the technique [19], where the consumer starts the reparation mechanism by activating the local functional test. Once it completes, the test result is sent back to the consumer for analysis. If a failure occurs, the consumer will send the repair message to the sensor node and initialize the backup component. Acknowledgement is sent back to the consumer once the reparation is completed. If the message latency from the consumer to the target node is d seconds and the test time is c seconds, then we calculate MTTR as Equation (5):

$$\text{Mean\_time\_to\_repair} : 4d + c \quad (5)$$

For the sensor node without the Test Interface Module [19], consumer sends the measured data request command to the suspected sensor node. In order to check the data integrity, same request command will also send to at least two other nearby sensor nodes. The consumer compares the three collected streams of data and pinpoints the failed node. Once the failure is confirmed, consumer will notify the surrounding sensor node to take over the applications of the failed node. Once the failure is confirmed, consumer will notify the surrounding sensor node to take over the applications of the failed node. Again if the message latency from the consumer to the target node is d seconds, then MTTR is:

$$\text{Mean\_time\_to\_repair} \sim 8d \quad (6)$$

To estimate realistic MTTR numbers, we use study [20], where for WSNs Thermostat application with 64 sensor nodes is simulated. Due to the power and protocol requirements, the average latency of related messages is 1522s. By applying this to our MTTR estimations, the test time c is much smaller and can be neglected.

Reliability of a system is defined as the probability of system survival Equation (7) in a period of time. Therefore, using Poisson probability [21] implemented for WSNs we have as well estimate probability of “failed” situation for whole WSN in given time interval, e.g. for one day (24 hours) to demonstrate the reliability of our presented approach.

$$\text{Probability}(r) = \frac{m^r \times e^{-m}}{r!} \quad (7)$$

Where Probability (r) is a probability of failure system working with “r” failed nodes within WSN for given time interval, $r \geq 0$, m is a average number of failed nodes within WSN and e = 2.718…

For example, in average there are 3 failed nodes in WSN fo 24 hours. Then we calculate Probabilities of failure system working as:

$$\text{Probability}(\text{"}r\text{" fails\_for\_24\_hours}) = \frac{3^r \times e^{-3}}{r!}$$

$$\text{Probability}\left(0\_\text{fail}\ \_\text{for}\ \_24\_\ \text{hours}\right)$$

$$= P(0) = \frac{3^0 \times e^{-3}}{0!} = 0.0498$$

$$\text{Probability}\left(1\_\ \text{fail}\ \_\ \text{for}\ \_24\_\ \text{hours}\right)$$

$$= P(1) = \frac{3^1 \times e^{-3}}{1!} = 0.1494$$

$$\text{Probability}\left(4\_\ \text{fails}\ \_\ \text{for}\ \_24\_\ \text{hours}\right)$$

$$= P(4) = \frac{3^4 \times e^{-3}}{4!} = 0.1680$$

From this example, we can see that with progressive increase of fail nodes quantity of a WSN, the risk of unstable work also increases.

5. Experiments and Evaluation

The discussion in this section will be about achieving two primary factors of dependability in WSNs applications, namely availability and reliability. In the classical definition, a system is highly available if the fraction of its downtime is very small, either because failures are rare, or because it can restart very quickly after a failure [22].

The performance of the proposed approach is demonstrated throughout using numerical examples. Reliability of a system is defined as the probability of system survival in a period of time. Since it depends mainly on the operating conditions and operating time, the metrics of Mean Time between Failure (MTBF) is used. For time period of duration t, MTBF is related to the reliability as follows [19]:

$$\text{Mean\_time\_between\_failure} \quad (8)$$

Availability of a system is closely related to the reliability, since it is defined as the probability that the system is operating correctly at a given time. Dependence availability and reliability on MTBF presented on Figure 2. Calculating availability is related to MTBF and Mean Time to Repair (MTTR) by the following relation [19]:

$$\text{Availability} = \frac{\text{Mean\_time\_between\_failure}}{\text{Mean\_time\_between\_failure} + \text{Mean\_time\_to\_repair}} \quad (9)$$

Considering availability of each node in isolation, from Equation (9), the MTTR should be minimized, while MTBF should be maximized. While MTBF is given by manufacturing practices and components used, the value of MTTR can be controlled by both individual node and network design.

$$M\ \% = \frac{m \times 100\%}{n} \quad (10)$$

where m is a number of failed nodes within WSN, n is number of nodes within WSN and $M\%$ is possible percentage of failed nodes within given WSN.

Serviceability of a system is defined as the probability that a failed system will restore to the correct operation. Serviceability is closely related to the repair rate and the MTTR [19].

$$\text{Serviceability} = 1 - \exp \times \left(\frac{t}{\text{Mean\_time\_to\_repair}} \right) \quad (11)$$

A fundamental service in sensor networks is the determination of time and location of events in the real world. This task is complicated by various challenging characteristics of sensor networks, such as their large scale, high network dynamics, restricted resources, and restricted energy. We use Hawk sensor nodes for determination time of data transmitting in fulfilling the QoS under these constraints. We illustrate the practical feasibility to our approaches by concrete application of real sensor nodes (Hawk Sensor Nodes) to our experiments and the results of availability and reliability of sensor nodes to reveal QoS from our experiment can be seen on Figure 2 above.

In any system one must consider the reliability of its components when ascertaining overall system performance. Thus our question was whether the proposed strategy performed adequately for various levels of sensor reliability. Equation (2), does not include any information regarding expected sensor life and thus assumes static network resources, which is clearly not the case in WSNs. For example, sensors may fail at regular intervals due to low reliability, due to cost driven design choices, environmentally caused effects (especially in harsh environments), loss of energy, etc.

We measured the processing throughput, i.e., the number of data transmitted events that each phase is able to process per second and time taken to transmit these data within selected sensor nodes, as can be seen in graph presentation in Figure 3.

We plot the node availability versus average latency, which lumps together the characteristics of the channel, the number of retransmission retries on the failure, as well as the node-dependent features such as retransmission timeouts

6. Explain about the Synchronization in WSN [CO5-L1]

The traditional clock synchronization protocols surveyed in the previous section are widely used in wired networks. However, they are not suitable for wireless sensor networks for a variety of reasons that we discuss in this section. Clock synchronization in wireless sensor networks requires newer and more robust approaches. A thorough understanding of the challenges posed by wireless sensor networks is crucial for the successful design of synchronization protocols for such networks. This section examines the design principles for clock synchronization in wireless sensor networks, and then classifies various such synchronization protocols. 3.1 Challenges of sensor networking Wireless sensor networks have tremendous potential because they will expand our ability to monitor and interact remotely with the physical world. Smart sensors have the ability to collect vast amounts of hitherto unknown data, which will pave the way for a new breed of computing applications as we showed in Table 1. Sensors can be accessed remotely and placed where it is impractical to deploy data and power lines. Nodes can be spaced closely, yielding finegrained pictures of real-world phenomena that are currently modeled only on a large scale. However, to exploit the full potential of sensor networks, we must first address the peculiar limitations of these networks and the resulting technical issues. Evidently, sensor networks can be best exploited by applications that perform data fusion to synthesize global knowledge from raw data on the fly. Although data fusion requires that nodes be synchronized, the synchronization protocols for sensor networks must address the following features of these networks.

1 Limited energy
While the efficiency of computing devices is increasing rapidly, the energy consumption of a wireless sensor network is becoming a bottleneck. Due to the small size and cheap availability of the sensors, sensor networks can employ thousands of sensors. This makes it impossible to wire each of these sensors to a power source. Also, the need for unmanned operation dictates that the sensors be battery-powered. Since the amount of energy available to such sensors is quite modest, synchronization must be achieved while preserving energy to utilize these sensors in an efficient fashion

.2 Limited bandwidth
In wireless sensor nets, much less power is consumed in processing data than transmitting it. Presently, wireless communication is restricted to a data rate in the order of 10–100 Kbits/second [21]. Pottie and Kaiser [55] have shown that the energy required to transmit 1 bit over 100 meters, which is 3 joules, can be used to execute 3

million instructions. Bandwidth limitation directly affects message exchanges among sensors, and synchronization is impossible without message exchanges.

3 Limited hardware

The hardware of a sensor node is usually very restricted due to its small size. A typical sensor node like the Berkeley Mica2 mote [35] has a small solar battery, an 8-bit CPU that runs at a speed of 10MHz, 128KB to 1MB memory, and a communication range of less than 50 meters. Hill et al. surveyed some sensor network platforms as well as the most popular sensor architectures, such as Spec, Smartdust, Intel's Imote [32], and Stargate. Figure 13 illustrates the configuration of a typical sensor node. The restrictions on computational power and storage space pose a huge challenge. The size of a sensor cannot be increased because it would make it more expensive and consume more power. This would prevent the deployment of thousands of sensor nodes, which is usually required for efficient operation of several critical applications. Transceiver Embedded Processor Battery Memory Sensors 66% of Total Cost Requires Supervision 1 Kbps−1Mbps, 3−100 meters Lossy Transmissions 8−bit, 10 MHz, Slow Computations Limited Lifetime Limited Storage 128KB−1MB Figure 13: Sensor node hardware for Mica mote

4. network connections

An implicit advantage usually available to a wireless network is mobility. Mobile ad-hoc networks are becoming increasingly popular and the following issues must be addressed. • The communication range of the mobile sensors is very limited (roughly 20–100 metres), which makes message exchanges between sensor nodes difficult. • A wireless medium is unshielded to external interference and this may lead to a high percentage of message loss. • A wireless connection suffers from a restricted bandwidth and intermittent connectivity. • The network topology frequently changes due to the mobility of the nodes. Dynamic reconfiguration becomes necessary

.

.5 Tight coupling between sensors and physical world

WSNs seek to monitor real-world phenomena and the network design is tailored to the specific environment being sensed. Therefore, as WSNs are used for critical and diverse applications like military tracking, forest fire monitoring, and geographical surveillance, the network has to be tailored to suit the application. For instance, sensors can be used to measure temperature, light, sound, or humidity, and the application (e.g., forest fire monitoring) decides the type of sensors to be used (e.g., temperature sensors). 3.2 Design principles of clock synchronization in sensor networks Researchers have developed a wide variety of clock synchronization protocols for traditional wired networks over the past few decades, as surveyed in Section 2. However, due to the peculiar characteristics, limitations, and the dynamic nature of wireless sensor networks, as seen in Section 3.1, these protocols cannot be applied directly. Several important design considerations are listed next.

1 Energy efficiency

• External time standard (GPS) usage In sensor nets, energy conservation is very important. Traditional protocols like NTP and TEMPO use an external standard like GPS (Global Positioning System) or UTC (Universal Time) to synchronize the network to an accurate time source. However, the use of GPS poses a high demand for energy which is usually not available in sensor networks. This makes it difficult to maintain a common notion of time. • Mode of transmission Reduction of energy is achieved by choosing to transmit over multiple short distances instead of a single long path. This translates into either a lower transmit power or a higher data transmission speed over a given distance. Either one will decrease the total end-to-end energy needed to transmit a packet of data. This implies that in large sensor networks, data is transmitted in sequences or hops, instead of a single long path from the sender to the receiver. • Proactive versus reactive routing A proactive protocol keeps track of all the nodes in a node's neighborhood, having total knowledge of all possible routes at all times. Reactive protocols do not maintain routing information proactively and find routes only when they need them. A reactive protocol leads to energy savings because nodes do not waste energy by attempting to maintain synchronization at all times. Nodes are awakened only when they are needed. Elson et al.'s Reference Broadcast Synchronization (RBS) uses a similar technique, called post-facto synchronization.

2 Infrastructure In many critical sensor applications,

the network is deployed in an ad-hoc fashion. Ad-hoc networks are networks of mobile wireless sensors in which the mobile nodes constantly change their neighborhood and the configuration. This denies the convenience of having an infrastructure like NTP which has several layers of servers that provide an accurate source of time. In ad-hoc sensor networks, the nodes must cooperate to organize themselves into a network and resolve contention for the available bandwidth. These tasks become more complex if the number of nodes grows or if the relationship among nodes changes rapidly, for instance, because of mobility.

3 End-to-end latency

Traditional wired networks are fully connected networks in which the variability in the propagation and (intermediate) queuing delay is relatively small. In addition, any node can send a message directly to another node at any point in time. This implies a constant end-to-end delay throughout the network and provides a close approximation for the actual latency. Sensor nets may be large in size and have to deal both with mobility and wireless transmission over a shared medium. These features make it impractical to assume a single latency bound between the ends of the network. Sensor nets therefore need localization algorithms to reduce this latency error as well as the jitter, the unpredictable variation in transmission times. Also, protocols that assume a fully connected network cannot be applied to multi-hop sensor networks.

.4 Message loss and message delivery

Fault-tolerant algorithms for traditional wired networks handle message loss by sending extra messages. This ensures that every node participates in the synchronization, leading to better operation. Several protocols for wired networks employ the averaging method to compute the delay between two nodes, which is a critical aspect in maintaining synchronized time. Message loss handling and estimating message delay by averaging are not desirable in sensor nets because of the following reasons. • Transmission of every bit requires energy and multiple message transmissions to estimate average delays lead to higher energy requirements. 12 • Message delivery is very unreliable due to the dynamic nature of the network, the intermittent connectivity, and the limited communication range of each node. 3.2.5 Network dynamics A stationary sensor network, without any mobility, usually requires an initial set-up before beginning operation. However, if the application demands a higher population of nodes in a particular part of the network, the addition of extra nodes changes the neighborhood of each node and the configuration of the network. Dynamic sensor networks add further challenges because the nodes are mobile. Mobility directly leads to a frequent change in topology of the network. Hence, the protocols used for such networks, whether stationary or dynamic, must ensure self-configuration (by use of suitable neighborhood definition or leader election protocols) to achieve synchronization. 3.3 Classification of synchronization protocols Wireless sensor networking can be applied to a wide range of applications, from simple parking lot monitoring to safety-critical applications like earthquake detection. As most networks are very closely coupled to the application, the protocols used for synchronization differ from each other in some aspects and resemble each other in other aspects. We classify synchronization protocols based on two kinds of features. 1. Synchronization issues 2. Application-dependent features

1 Synchronization issues Wireless sensor networks provide answers to user queries by fusing data from each sensor to form a single answer or result. To accomplish this data fusion, it becomes necessary for these sensors to agree on a common notion of time. All the participating sensors can be enveloped in a common time scale by either synchronizing the local clocks in each sensor or by just translating timestamps that arrive at a sensor into the local clock times. Various options are now described. • Master-slave versus peer-to-peer synchronization Master-slave. A master-slave

protocol assigns one node as the master and the other nodes as slaves. The slave nodes consider the local clock reading of the master as the reference time and attempt to synchronize with the master. In general, the master node requires CPU resources proportional to the number of slaves, and nodes with powerful processors or lighter loads are assigned to be the master node. Mock et al. have adopted the IEEE 802.11 clock synchronization protocol due to its simple, non-redundant, master/slave structure. Ping's protocol also adheres to the master-slave mode. Peer-to-peer. Most protocols in the literature, such as RBS, Romer's protocol and the asynchronous diffusion protocol of Li and Rus are based on a peer-to-peer structure. Any node can communicate directly with every other node in the network. This eliminates the risk of master node failure, which would prevent further synchronization. Peer-to-peer configurations offer more flexibility but they are also more difficult to control. • Clock correction versus untethered clocks Clock correction. Most methods in practice perform synchronization by correcting the local clock in each node to run on par with a global time scale or an atomic clock, which is used to provide a convenient reference time. The protocol of Mock et al. [49] and Ping's protocol [54] are based on this method. The local clocks of nodes that participate in the network are corrected either instantaneously or continually to keep the entire network synchronized. 13 Untethered clocks. Achieving a common notion of time without synchronization is becoming popular, because a considerable amount of energy can be saved by this approach. RBS [builds a table of parameters that relate the local clock of each node to the local clock of every other node in the network. Local timestamps are then compared using the table. In this way, a global time scale is maintained while letting the clocks run untethered. Romer uses the same principle. When timestamps are exchanged between nodes, they are transformed to the local clock values of the receiving node. The round-trip delay between two nodes and the idle time of a message are taken into consideration. • Internal synchronization versus external synchronization Internal synchronization. In this approach, a global time base, called real-time, is not available from within the system and the goal is to minimize the maximum difference between the readings of local clocks of the sensors. The protocol of Mock et all uses internal synchronization. External Synchronization. In external synchronization, a standard source of time such as Universal Time (UTC) is provided. Here, we do not need a global time base since we have an atomic clock that provides actual real-world time, usually called reference time. The local clocks of sensors seek to adjust to this reference time in order to be synchronized. Protocols like NTP synchronize in this fashion because external synchronization is better suited to loosely coupled networks like the Internet. Most protocols in sensor networks do not perform external synchronization unless the application demands it, because energy efficiency is a primary concern and employing an external time source typically induces high-energy requirements. Internal synchronization usually leads to a more correct operation of the system, while external synchronization is primarily used to give users convenient reference time information. Note that internal synchronization can be performed in a master-slave or peer-to-peer fashion. External synchronization cannot be performed in a peer-to-peer fashion; it requires a master node which communicates with a time service like GPS to synchronize the slaves and itself to the reference time. • Probabilistic versus deterministic synchronization Probabilistic synchronization. This technique provides a probabilistic guarantee on the maximum clock offset with a failure probability that can be bounded or determined. The reasoning behind a probabilistic approach is that a deterministic approach usually forces the synchronization protocol to perform more message transfers and induces extra processing. In a wireless environment where energy is scarce, this can be very expensive. The protocol of PalChaudhuri is a probabilistic variation of RBS defined a probabilistic protocol for wired networks. Deterministic synchronization. Arvind defines deterministic algorithms as those that guarantee an upper bound on the clock offset with certainty. Most algorithms in the literature are deterministic. Sichitiu and Veerarittiphan's protocol is centered on a deterministic algorithm. RBS and the time-diffusion protocol] are also deterministic. • Sender-to-receiver versus receiver-to-receiver synchronization Most existing methods synchronize a sender with a receiver by transmitting the current clock values as timestamps. As a consequence, these methods are vulnerable to variance in message delay. Newer methods such as RBS perform synchronization among receivers using the time at which each receiver receives the same message. Such an approach reduces the time-critical path, which is the path of a message that contributes to non-deterministic errors in the protocol. Sender-to-receiver synchronization. This traditional approach usually happens in three steps. 1. The sender node periodically sends a message with its local time as a timestamp to the receiver. 2. The receiver then synchronizes with the sender using the timestamp it receives from the sender. 3. The message delay between the sender and receiver is calculated by measuring the total round-trip time, from the time a receiver requests a timestamp until the time it actually receives a response. 14 The drawbacks of this approach are obvious. There is a variance in message delay between the sender and the receiver. The variance is due to network delays (prominent in multi-hop networks) and the workload in the nodes that are involved. Most methods compute the average message delay after performing many trials, during which they lose accuracy and add further overhead. Also, optimization of the time taken by the sender to prepare and transmit the message, and the time taken by the receiver to process the message must be considered. Receiver-to-receiver synchronization. This approach exploits the property of the physical broadcast medium that if any two receivers receive the same message in single-hop transmission (see below), they receive it at approximately the same time. Instead of interacting with a sender, receivers exchange the time at which they received the same message and compute their offset based on the difference in reception times. The obvious advantage is the reduction of the message-delay variance. This protocol is vulnerable only to the propagation delay to the various receivers and the differences in receive time. Table 2 classifies the various protocols for clock synchronization, based on the analysis in this section. SYNCHRONIZATION ISSUES Protocol Master-slave vs. Internal vs. Probabilistic vs. Sender-to-receiver vs. Clock Peer-to-Peer External Deterministic Receiver-to-receiver Correction RBS Peer-to-peer Both Deterministic Receiver-to-receiver No Romer Peer-to-peer Internal Deterministic Sender-to-receiver No Mock et alMaster/slave Internal Deterministic Receiver-to-receiver Yes Ganeriwal et al. Master/slave Both Deterministic Sender-to-receiver Yes Ping Master/slave Both Deterministic Sender-to-receiver Yes PalChaudhuri et al. Peer-to-peer Internal Deterministic Sender-to-receiver Yes Time-diffusion protocol Peer-to-peer Internal Deterministic Receiver-to-receiver Yes Asynchronous diffusion Peer-to-peer Internal Deterministic Sender-to-receiver Yes Table 2: Classification based on synchronization issues.

2 Application-dependent features • Single-hop versus multi-hop networks Single-hop communication. In a single-hop network, a sensor node can directly communicate and exchange messages with any other sensor in the network. However, many wireless sensor network applications span several domains or neighborhoods. (Nodes within a neighborhood can communicate via singlehop message transmission.) The network is often too large, making it impossible for each sensor node to directly exchange messages with every other node. Elson and show that a single latency bound cannot be assumed. Protocols such as those by Mock et alGaneriwal et al. [ing and PalChaudhuri et alare based on single-hop communication; however, they can be extended to multi-hop communication. Multi-hop communication. The need for multi-hop communication arises due to the increase in the size of wireless sensor networks. In such settings, sensors in one domain communicate with sensors in another domain via an intermediate sensor that can relate to both domains Communication can also occur as a sequence of hops through a chain of pairwise-adjacent sensors. RBS Ping's protocol the protocol by PalChaudhuri et al. and Su and Akyildiz's time-diffusion protocol can be suitably extended to handle multi-hop communication. • Stationary networks versus mobile networks Most sensor networks are tightly coupled to the application and synchronization protocols vary depending on the application at hand. Mobility is an inherent advantage of a wireless environment but it induces more difficulties 15 in achieving synchronization. It leads to frequent changes in network topology and demands that the protocol be more robust. Stationary networks. In stationary sensor networks, the sensors do not move. An example is a network of sensors for monitoring the motion of a vehicle in a certain area. For these sensor networks, the topology remains unchanged once the sensors are deployed in the environment. The protocols used by Mock et al. Ganeriwal et al. and PalChaudhuri et al. are geared to stationary networks. Mobile networks. In a mobile network, the sensors have the ability to move,

and they connect with other sensors only when entering the geographical scope of those sensors. The scope of a mobile sensor is the communication range up to which it can communicate and successfully exchange messages with other sensors. Romer shows the need for a robust protocol, which can handle the frequent changes in network topology due to the mobility of the nodes. The change in topology is often a problem because it requires resynchronization of nodes and recomputation of the neighborhoods or clusters. • MAC-layer based approach versus standard approach The Media Access Control (MAC) layer is a part of the Data Link Layer of the Open System Interconnection (OSI) model. This layer is responsible for the following functions. – Providing reliability to the layers above it with respect to the connections established by the physical layer. – Preventing transmission collisions so that the message transmission between one sender and the intended receiver node(s) does not interfere with transmission by other nodes. MAC protocols effectively utilize the MAC layer to achieve better energy efficiency, which is crucial in sensor networks. The IEEE 802.11 MAC protocol is widely used. Several variations of this protocol have been defined for the purpose of controlling power consumption, including the protocols listed below. A survey by Chen et al. compares some of these protocols. – S-MAC (Sensor-MAC) – PAMAS (Power-Aware Multi-Access Protocol) – EC-MAC (Energy-Conserving MAC) – PRMA (Packet-Reservation Multiple Access MAC) – DQRUMA (Distributed-Queuing Request Update Multiple Access) – MDR-TDMA (Multiservice Dynamic Reservation TDMA) . Reference Broadcast Synchronization does not rely on MAC protocols in order to avoid a tight integration of the application with the MAC layer. The protocols used by Mock et al., Ganeriwal et al, and Sichitiu and Veerarittipha rely on the CSMA/CA protocol for the MAC layer. A survey of MAC protocols for sensor networks is given by Jones et al. . Table 3 classifies the various protocols for clock synchronization, based on the analysis in this section.